

CONTROL-M for z/OS® User Guide



Supporting

Version 6.3.04 of CONTROL-M for z/OS

March 2009



Contacting BMC Software

You can access the BMC Software website at http://www.bmc.com. From this website, you can obtain information about the company, its products, corporate offices, special events, and career opportunities.

United States and Canada

Address BMC SOFTWARE INC Telephone 713 918 8800 or Fax 713 918 8000

800 841 2031

2101 CITYWEST BLVD HOUSTON TX 77042-2827

USA

Outside United States and Canada

Copyright 2009 BMC Software, Inc., as an unpublished work. All rights reserved.

BMC, BMC Software, and the BMC Software logo are the exclusive properties of BMC Software, Inc., are registered with the U.S. Patent and Trademark Office, and may be registered or pending registration in other countries. All other BMC trademarks, service marks, and logos may be registered or pending registration in the U.S. or in other countries. All other trademarks or registered trademarks are the property of their respective owners.

DB2 is a registered trademark of International Business Machines Corporation.

BMC Software considers information included in this documentation to be proprietary and confidential. Your use of this information is subject to the terms and conditions of the applicable End User License Agreement for the product and the proprietary and restricted rights notices included in this documentation.

Restricted rights legend

U.S. Government Restricted Rights to Computer Software. UNPUBLISHED -- RIGHTS RESERVED UNDER THE COPYRIGHT LAWS OF THE UNITED STATES. Use, duplication, or disclosure of any data and computer software by the U.S. Government is subject to restrictions, as applicable, set forth in FAR Section 52.227-14, DFARS 252.227-7013, DFARS 252.227-7014, DFARS 252.227-7015, and DFARS 252.227-7025, as amended from time to time. Contractor/Manufacturer is BMC SOFTWARE INC, 2101 CITYWEST BLVD, HOUSTON TX 77042-2827, USA. Any contract notices should be sent to this address.

Customer support

You can obtain technical support by using the Support page on the BMC Software website or by contacting Customer Support by telephone or e-mail. To expedite your inquiry, please see "Before Contacting BMC Software."

Support website

You can obtain technical support from BMC Software 24 hours a day, 7 days a week at http://www.bmc.com/support_home. From this website, you can

- read overviews about support services and programs that BMC Software offers
- find the most current information about BMC Software products
- search a database for problems similar to yours and possible solutions
- order or download product documentation
- report a problem or ask a question
- subscribe to receive e-mail notices when new product versions are released
- find worldwide BMC Software support center locations and contact information, including e-mail addresses, fax numbers, and telephone numbers

Support by telephone or e-mail

In the United States and Canada, if you need technical support and do not have access to the web, call 800 537 1813 or send an e-mail message to support@bmc.com. Outside the United States and Canada, contact your local support center for assistance.

Before contacting BMC Software

Before you contact BMC Software, have the following information available so that Customer Support can begin working on your problem immediately:

- product information
 - product name
 - product version (release number)
 - license number and password (trial or permanent)
- operating system and environment information
 - machine type
 - operating system type, version, and service pack or other maintenance level such as PUT or PTF
 - system hardware configuration
 - serial numbers
 - related software (database, application, and communication) including type, version, and service pack or maintenance level
- sequence of events leading to the problem
- commands and options that you used
- messages received (and the time and date that you received them)
 - product error messages
 - messages from the operating system, such as file system full
 - messages from related software

License key and password information

If you have questions about your license key or password, contact BMC as follows:

- (USA or Canada) Contact the Order Services Password Team at 800 841 2031, or send an e-mail message to ContractsPasswordAdministration@bmc.com.
- (Europe, the Middle East, and Africa) Fax your questions to EMEA Contracts Administration at +31 20 354 8702, or send an e-mail message to password@bmc.com.
- (Asia-Pacific) Contact your BMC sales representative or your local BMC office.



Contents

About This Guide	33
Conventions Used in This Guide	34
Information New to This Version	
Information Relating to CONTROL-M/Restart Users	
Related Publications	
Chapter 1 Introduction to CONTROL-M	41
INCONTROL Products and IOA	42
IOA	42
INCONTROL	42
Functional Approach	43
Main Components	
Expanded CONTROL-M Functionality	
CONTROL-M Support for MAINVIEW Batch Optimizer	
Online User Interface to CONTROL-M	
CONTROL-M Concepts	
IOA Core and CONTROL-M Repository	62
Date Definition Concepts	63
Date Standards and Date Field Formats	65
Job Ordering and Job Forcing	
Rerun and Restart	67
Order ID	68
SYSDATA	68
Handling of Job Groups	68
Prerequisite Conditions	69
Quantitative and Control Resources	
Job Priority	
Automatic Job Flow Adjustment	
Chapter 2 Online Facilities	77
Overview	
IOA Features	
General IOA Features	
IOA Entry Panel	
IOA Primary Option Menu	
Multi-Screen Control.	
Fast Exit from the IOA Online Facility	
Screen Layout	92

Commands and PFKeys	93
Online Help	99
AutoRefresh Mode	100
IOA Under ISPF	101
IOA Editor	102
IOA SET Command Panel	
IOA TSO Command Processor Screen	108
Scheduling Definition Facility	110
Entry Panel	
Table List screen	
Job List Screen	
Job Scheduling Definition Screen – Defining Schedules	127
Exiting the Scheduling Definition Facility	
Ordering (Scheduling) Jobs	
Copying Jobs to Another Table	
Deleting Tables	
Condition Inheritance	
Displaying Graphic Jobflow	
Displaying a Job Scheduling Plan	
Tracking and Control Facility	166
Active Environment Screen	
Global View Screen	199
View Graph Screen	
Why screen	
Deleting a Job	
Log Screen	
Zoom Screen	
Confirm Scheduling Window	
Confirm Rerun Window	
§Restart§ Confirm Restart Window (Under CONTROL-M/Restart)	
§Restart§Rerun and/or Restart Window (Under CONTROL-M/Restar	
Rerun Flow Job List Window	
Step List Window	233
§Restart§Job Order Execution History Screen	
§Restart§ Sysout Viewing Screen	
Statistics Screen	
Job Dependency Network Screen	
History Environment Screen	
Force OK Confirmation Window	
CMEM Rule Definition Facility	249
Entry Panel	
Table List Screen	
Rule List Screen	
Rule Definition Screen - Defining Rules	
Entering Comments	
Editing CMEM Rule Definitions in the Edit Environment	
Exiting the CMEM Rule Definition Facility	
Deleting Tables	
Ordering CMEM Rule Tables	

Copying Rules to Another Table	
IOA Variables Database Facility	
Entry Panel	
Database List Screen	
Values of Database Screen	
Variable Zoom Screen	
Condition and Resource Handling Facility	281
IOA Conditions/Resources Screen	
IOA Manual Conditions Screen	291
IOA Log Facility	296
IOA Log Screen	296
IOA Calendar Facility	306
Accessing the IOA Calendar Facility	307
Entry Panel	308
Calendar List Screen	309
Year List Screen	310
Copying Years to Another Calendar	314
Calendar Definition Screen	
Exiting the IOA Calendar Facility	
Utilities Under ISPF	
IOA Online Utilities Menu.	
I1: Add, Check, or Delete a Prerequisite Condition	
M1: Issue a Job Order	
M2: Perform an AutoEdit Simulation	
M3: Prepare Simulation/Tape Pull List Job	
M4: Parameter Prompting Facilities	
M5: Quick Schedule Definition	
M6: End-User Job Order Interface.	
U1: Invoke DOCU/TEXT.	
OI. HIVORE DOCO/ TEXT	312
Chapter 3 Job Production Parameters	373
	~~~
Overview	
General Parameters – Summary	
Basic Scheduling Parameters – Summary	
Runtime Scheduling Parameters – Summary	
Post-Processing Parameters – Summary	
Parameter Descriptions	
ADJUST CONDITIONS: General Job Parameter	
APPL: General Job Parameter	
APPL FORM: General Job Parameter	395
APPL TYPE: General Job Parameter	396
APPL VER: General Job Parameter	397
§Restart§ AUTO-ARCHIVE: Post–Processing Parameter	398
CM VER: General Job Parameter	
CONFCAL: Basic Scheduling Parameter	
CONFIRM: Runtime Scheduling Parameter	
CONTROL: Runtime Scheduling Parameter	
CTB STEP: General Job Parameter.	
D-CAT: Basic Scheduling Parameter.	

DATES: Basic Scheduling Parameter	418
DAYS: Basic Scheduling Parameter	420
DEFINITION ACTIVE: Basic Scheduling Parameter	430
DESC: General Job Parameter	
DO Statement: Post-Processing Parameter	
DO COND: Post-Processing Parameter	
DO CTBRULE: Post-Processing Parameter	
DO FORCEJOB: Post-Processing Parameter	
§Restart§DO IFRERUN: Post-Processing Parameter	447
DO MAIL: Post-Processing Parameter	
DO NOTOK: Post-Processing Parameter	
DO OK: Post-Processing Parameter	
DO REMEDY: Post-Processing Parameter	
DO RERUN: Post-Processing Parameter	
DO SET: Post-Processing Parameter	
DO SHOUT: Post-Processing Parameter	
DO STOPCYCL: Post-Processing Parameter	
DO SYSOUT: Post-Processing Parameter	
DOC: General Job Parameter	
DOCLIB: General Job Parameter	
DOCMEM: General Job Parameter	
DUE OUT: Runtime Scheduling Parameter	
GROUP: General Job Parameter	
GRP MAXWAIT: Basic Scheduling Parameter	
IN: Runtime Scheduling Parameter	500
INSTREAM JCL: General Job Parameter	511
INTERVAL: Post-Processing Parameter	
MAXRERUN: Post-Processing Parameter	
MAXWAIT: Basic Scheduling Parameter	
MEMLIB: General Job Parameter	
MEMNAME: General Job Parameter	
MINIMUM: Basic Scheduling Parameter	
MONTHS: Basic Scheduling Parameter	
NJE NODE: General Job Parameter	
ON Statements: Post–Processing Parameter	537
ON GROUP-END: Post-Processing Parameter	
ON PGMST: Post-Processing Parameter	
ON SYSOUT: Post–Processing Parameter	
OUT: Post-Processing Parameter	
OVERLIB: General Job Parameter	
OWNER: General Job Parameter	
PDS: Basic Scheduling Parameter	
PIPE: General Job Parameter	
\$Restart\$PREVENT-NCT2:General Job Parameter	
PRIORITY: Runtime Scheduling Parameter	
RELATIONSHIP: Basic Scheduling Parameter	
RERUNMEM: Post-Processing Parameter	
RESOURCE: Runtime Scheduling Parameter	
RETENTION: # OF DAYS TO KEEP: Post-Processing Parameter	

RETENTION: # OF GENERATIONS TO KEEP: Post–Processing Parameter	605
RETRO: Basic Scheduling Parameter	607
SAC: Run Time Parameter	610
SCHEDULE TAG: Basic Scheduling Parameter	612
SCHEDULE TAG ACTIVE: Basic Scheduling Parameter	616
SCHENV: General Job Parameter	619
SET VAR: General Job Parameter	
SHOUT: Post-Processing Parameter	627
STAT CAL: General Job Parameter	639
STEP RANGE: Post-Processing Parameter	641
SYSOUT: Post-Processing Parameter	644
SYSTEM ID: General Job Parameter	652
TASKTYPE: General Job Parameter	654
TIME + DAYS: Runtime Scheduling Parameter	659
TIME ZONE: Runtime Scheduling Parameter	667
WDAYS: Basic Scheduling Parameter	
Chapter 4 CONTROL-M Event Manager (CMEM)	679
Overview	680
Types of Events Managed by CMEM	
Types of Actions That CMEM Can Perform	
CMEM Rule Ordering, Triggering and Deactivation	
CMEM AutoEdit Variables	
On Spool Jobs	
Support for ON DSNEVENT and ON STEP	
CMEM Support for FTP	
CMEM Support for IBM FTP	
Rule Parameters – Summary	
Event Selection Parameters	
General Parameters	
Action Parameters	
Parameter Descriptions	
DESCRIPTION: General Parameter	
DO statement: Action Parameter	
DO RULE: Action Parameter	
DO SHOUT: Action Parameter	
	717
	721
ON DSNEVENT: Event Parameter	
ON JOBARRIV: Event Parameter	
ON JOBEND: Event Parameter	
ON STEP: Event Parameter	
OWNER: General Parameter	
RUNTSEC: General Parameter	
THRESHOLD: Runtime Scheduling Parameter	

Chapter 5 JCL and AutoEdit Facility	743
Overview	745
System variables	748
Non-Date System Variables	748
Date System Variables	
Special System Variables	
User-Defined Variables	
Valid Characters in User-Defined Variables	755
Local Variables	
Global Variables	760
JCL Setup Operation Flow	765
Rules of Variable Resolution	
Order of Precedence for Multiple Value Assignments	
Control Statements	
%%GLOBAL	
%%GOTO and %%LABEL	
%%IF, %%ELSE, %%ENDIF	
%%INCLIB and %%INCMEM	
%%LIBSYM and %%MEMSYM	
%%RANGE.	
%%RESOLVE	
%%SET	
Operators	
Functions	
%%\$CALCDTE.	
%%\$GREG	
%%\$JULIAN	
%%SLEAP	
%%\$WCALC	
%%\$WEEK#	
%%\$WEEKDAY	
%%\$YEARWK#	
%%CALCDATE	
%%SUBSTR	788
%%\$LENGTH.	
%%\$TYPE	
%%\$FUNC	
Testing AutoEdit Syntax	
AutoEdit Usage in the Job Scheduling Definition	
Examples	
Date Variables.	
ODATE, RDATE and DATE Usage	
How to Obtain Date Formats – 1	
How to Obtain Date Formats – 2	
How to Obtain Date Formats – 2	
How to Obtain Date Formats – 3	
Automatic Job Order for the Next Day	
Tape Clearance System – Stage 1	
Tane Clearance System – Stage 2	

Tape Management System	803
Dynamic Job Name	
Controlling the Target Computer by Class	804
Controlling the Target Computer by System Affinity	
%%BLANKn Statement	
%%RANGE Statement	806
SYSIN Parameter Containing %%	
%%INCLIB and %%INCMEM Statements	
Boolean "IF" Logic	
Chapter 6 Selected Implementation Issues	811
Overview	
Job Ordering Methods	
Job Ordering Through Quick Submit Command CTMQSB	
Special Purpose Job Ordering From Special Environments: CTMAJO	
Manual Conditions File and Maybe Jobs	
Loading the Manual Conditions File	817
Using the Manual Conditions File	817
Handling Manual Conditions	818
Handling Unscheduled Conditions	
Handling Maybe Dependencies	819
Maybe Jobs in Group Scheduling Tables	
MAINVIEW Batch Optimizer Considerations	
Job-Related Considerations for Pipes	
Enhanced Runtime Scheduling Algorithm	
System-Related Considerations	
Parameter Prompting Facilities	
Parameter Prompting Facility – Type 1	
Summary	
Usage Notes	
Parameter Prompting Facility—Type 2	
Maintenance Utilities	
Maintenance Offices	გაგ
Chapter 7 Simulation and Forecasting Facility	839
Overview	840
Simulation Procedure CTMSIM	840
Activating the Procedure	
Preparatory Steps	
Parameters	
Input Files	
Output Files	
CONTROL-M Exits and Simulation Processing	
Analyzing the Simulation Run	
v	
Handling Manual Conditions	
The CTMTAPUL Tape Pull List Procedure	
Activating the Procedure	
Parameters	
DD Statements	855

JOB/SCAN, PRO/JCL, DOCU/TEXT Interface	. 855
Problem Determination for Tape Pull List Reports	
Sample Tape Pull List Reports	
Appendix A The CONTROL-M Application Program Interface (CTMAPI)	861
Overview	862
Environment and Allocations.	
Functions	
1. Order or Force Existing Jobs	
Order or Force under Batch, REXX or CLIST	. 866
Order or Force Using Program	
Order or Force Allocations	
Order or Force Return Codes	
Order or Force Performance Considerations	. 868
Order or Force Security Considerations	
2. Create, Order, or Force New Tables.	
Invoking Create, Order or Force New Tables Using Program	. 868
Create, Order or Force Allocations	. 869
Create, Order or Force Return Codes	
Create, Order or Force Performance Considerations	
Create, Order or Force Security Considerations	
3. AJF Actions	
AJF Action under Batch, REXX or CLIST	. 870
AJF Action using Program	
AJF Action Allocations	. 871
AJF Action Return Codes	. 871
AJF Action Performance Considerations	. 872
AJF Action Security Considerations	. 872
4. Search	
Search under Batch, REXX or CLIST	. 873
Invoking Search from a Program	
Search Allocations	
Search Return Codes	
Search Performance Considerations	
Search Security Considerations	
5. Global Variables	
Global Variables under Batch REXX or CLIST	
6. Conditions and Resources	
7. CTMAS Active Job Download Filtering	. 876
EMDOWNLD Action under Batch, REXX or CLIST	
EMDOWNLD Action Allocations	
EMDOWNLD Action Return Codes	
EMDOWNLD Action Performance Considerations	
EMDOWNLD Action Security Considerations	
Examples	. 878
Conditional Requests and Selection Criteria	
Performance Considerations for Selection Criteria	
Search Security Considerations	. 881 882
RATHER LAGAC	xx,

Conversational Mode using Program	882
Input and Output Registers	
Example – Invoking CTMAPI in BAPI mode	
CTMBAPI DSECT	
Status Extension	887
The Status Reply DSECT (CTMBJSE)	889
Status Allocations	
Status Return Codes	891
Status Performance Considerations	891
Status Security Considerations	892
Order Extension	
Order Return Codes	
Order Reply	
Order or Force Allocations	
Order or Force Security Consideration	
AJF Action Extension.	
Identifying the Job	
Defining the Action	
Action Return Codes	
Action AJF Allocations	
Action Security Considerations	
Global Variable Extension	
Global Variable Return Codes	
Quantitative Resource Extension.	
Quantitative Resource Return Codes	
Quantitative Resource Security Considerations	
Quantitative Resource Allocations	
Create and/or Order or Force a Table (BLT)	
BLT Action Return Codes	
BLT Reply	
BLT Security Considerations	
BLT Resource Allocations.	
Replies	
CTMBAPO	
Date Format Considerations	
Date Format Considerations	903
Appendix B CONTROL-M for z/OS Unix System Services (USS)	905
Implementation Options	905
OS/390-Oriented Implementation	
Unix Oriented Implementation	
Integrating SAP R/3 running on USS	908
CONTROL-M Support for SAP in the USS Environment	909
11	
Appendix C Editing Job Scheduling Definitions in the Edit Environment	913
Line Editing Commands	915
Maintaining Valid Job Scheduling Definitions	917

Appendix D	Editing CMEM Rule Definitions in the Edit Environment	927
	Commands	
Appendix E	MVS Job Restart Without CONTROL-M/Restart	933
Index		937



# **Figures**

Establishing Job Dependency by Prerequisite Conditions	70
IOA Entry Panel	85
IOA Primary Option Menu where only CONTROL-M is Installed	86
IOA Primary Option Menu when all INCONTROL Products are Installed	88
IOA Version Information	90
IOA Log Screen	92
PFKey Assignment Window	94
IOA Help Screen	99
IOA Editor Edit Entry Panel 1	02
IOA Editor 1	
IOA SET Command Panel 1	06
IOA TSO Command Processor Screen	
CONTROL-M Scheduling Definition Facility - Entry Panel	
CONTROL-M Scheduling Definition Facility - Entry Panel Search Window 1	
CONTROL-M Scheduling Definition Facility Table List Screen 1	
CONTROL-M Scheduling Definition Facility Job List Screen	
Job Scheduling Definition Screen	
General Job Parameters 1	
Basic Scheduling Parameters - Job	
Basic Scheduling Parameters - Group Entity 1	
Runtime Scheduling Parameters	
Post-Processing Parameters	
Group Entity Scheduling Definition Screen 1	39
Editing Job Scheduling Definitions in the Edit Environment Screen	
Job Scheduling Definition DOC lines	
Save Documentation Window 1	
Job List Screen Exit Option Window	
Order and Force Confirmation Window (Groups)	
Order and Force Confirmation Window (Jobs)	
Select Group window	
The Double Confirmation Window	
Window for Copying Jobs to Another Table	
Delete Table Confirmation Window	
Condition Inheritance Confirmation Window	
Graphic Jobflow for Color Terminals	
Graphic Jobflow for Non-Color Terminals	
Job Scheduling Plan Window	
Job Scheduling Plan Screen	
CONTROL-M Active Environment Screen	
Display Type D (Default)	71

Display Type A (All Fields)	173
Display Filters Window	
Show Screen Filter Window	192
Global View Screen	
View Graph Screen Format for Color Terminals	202
View Graph Screen Format for Non-Color Terminals	
Active Environment Why Screen	205
Why Screen Add Condition or Delete NOT-COND Confirmation Window	209
Active Environment Screen Delete Confirmation Window	
Active Messages Log Screen	
CONTROL-M Zoom Screen	
Zoom Screen for Group Entities	
Adding or Editing a Job Order Note	
Exiting the Zoom Screen Confirmation Window	
Active Environment Screen Confirm Rerun Window	
<b>§Restart§</b> Active Environment Rerun and/or Restart Confirmation Window	
§Restart§ Flow Rerun Options Window	
Rerun Flow Job List Window	
Rerun Flow Job List Confirmation Window	
Rerun and/or Restart Step List Window	
<b>§Restart§</b> Job Order Execution History Screen	
<b>§Restart§</b> Sysout Viewing Screen	
Active Environment Statistics Screen	
Tape Device Usage Statistics	
Job Dependency Network Display Type N (Network)	
History Environment Screen	246
CONTROL-M Active Environment FORCE OK confirmation window	
CMEM Rule Definition Facility – Entry Panel	
CMEM Definition Facility Table List Screen	
CMEM Rule Definition Rule List Screen	
Rule Definition Screen - Defining Rules	
CMEM Rule Definition Event Selection Parameters - Example	
General Parameters	
Rule Definition Screen Comment Usage	
Rule List Screen Exit Option Window	
Rule Definition Facility Delete Table Confirmation Window	
Order and Force Confirmation Window	
Window for Copying Rules to Another Table	
IOA Variable Database Entry Panel	
IOA Variable Database Facility Database List Screen	
Variable Zoom Screen	
IOA Conditions/Resources Screen	
IOA Conditions/Resources COND Window	
IOA Conditions/Resources DELETE Confirmation Window	
IOA Conditions/Resources CHANGE Option Window	
Resource Analysis WHY Option Window	
IOA Manual Conditions Screen NEW Window	
10// IVIGINAL COMUNICIS SUITEM INDIVINUOVI	

IOA Manual Conditions Screen ERASE Confirmation Window	295
IOA Log Screen	
IOA Log Screen Display Filters Window	300
IOA Log Show Screen Window	
IOA Log Show Screen Window at Sites where Multiple INCONTROL Products	are
Active	305
IOA Calendar Facility - Entry Panel	308
Calendar List Screen	
Year List Screen	311
Calendar List Screen Copy Window	314
Calendar Definition Screen	315
Use of Reserved String "==PERIODIC=="	317
Periodic Calendar – Example 1	
Periodic Calendar – Example 2	318
Calendar List Screen Delete Confirmation Window	320
Year List Screen Exit Option Window	322
IOA Online Utilities Menu when all INCONTROL Products are Installed	
Prerequisite Condition Utility Screen	325
Job Request Utility Screen	
CONTROL-M AutoEdit Simulation Screen	
CONTROL-M Simulation and Forecasting Facility and Tape Pull List	333
Parameters Prompting Entry Panel	
Parameter Prompting Facility (Type 1) Primary Menu	
Define Parameters and Condition - New Master Table Screen	
Define Parameters/ and Conditions - Master Table Definition Screen	
Define Parameters and Conditions Screen	
Define Parameters and Conditions Save Screen Window	
Update Parameters and Set conditions - Table Selection Screen	
Table Selection Screen Delete Confirmation Window	
Update Parameters and Set Conditions Screen	
Update Parameters and Set Conditions - Confirm Parameter Update Actions	
Parameter Prompting Facility (Type 2) Primary Menu	
Primary Prompting Facility – Define or Update a Master Plan	
Parameter Prompting Facility – Master Plan Definition	
Define Parameters in the Master Plan Screen	
Fetch a Plan Screen	
Exec/Order a Plan (CTMEXEC) Screen	
Plan Selection Screen	
Update Parameters Values Screen	
CONTROL-M Quick Schedule Definition Screen	
CONTROL-M Quick Search Schedule Definition	
Quick Schedule Definition Job List Screen Entered	
Quick Schedule Definition Facility Exit Option Window	
Scheduling Definition Screen Quick Schedule Definition Example	
Job List Screen Entered Through the End-User Job Order Interface	
Job Scheduling Date and FORCE Options Window	
Job Scheduling Definition Screen	
Group Entity Definition Screen	
Group Scheduling Flowchart	

ADJUST CONDITIONS Parameter Format	. 389
ADJUST CONDITIONS Parameter Example	. 391
APPL Parameter Format	. 393
APPL Parameter Example	. 394
APPL FORM Parameter Format	. 395
APPL TYPE Parameter Format	. 396
APPL VER Parameter Format	. 397
<b>§Restart§</b> AUTO-ARCHIVE Parameter Format	. 398
<b>§Restart§</b> AUTO-ARCHIVE Parameter Example	. 400
CM VER Parameter Format	
CONFCAL Parameter Format	. 402
Days When Job Scheduled	. 405
CONFIRM Parameter Format	. 406
CONFIRM Parameter Example	. 407
CONTROL Parameter Format	. 408
CONTROL Parameter Example 1	. 411
CONTROL Parameter Example 2	. 411
CONTROL Parameter Example 3	. 412
CTB STEP Parameter Format	
CTB STEP Parameter Example	. 415
DCAT Parameter Format	. 416
DCAT Parameter Example	. 417
DATES Parameter Format	. 418
DATES Parameter Example	. 419
DAYS Parameter Format	. 420
DAYS Parameter Example 1	. 425
DAYS Parameter Example 2	. 425
DAYS Parameter Example 3	. 426
DAYS Parameter Example 4	. 426
DAYS Parameter Example 5	. 426
DAYS Parameter Example 6	. 427
DAYS Parameter Example 7	. 427
DAYS Parameter Example 8	. 428
DAYS Parameter Example 9	. 428
DAYS Parameter Example 10	. 429
DEFINITION ACTIVE Parameter Format	
DESC Parameter Format	. 432
DESC Parameter Example	. 433
DO Parameter Format	. 434
DO COND Parameter Format	. 436
Long DO COND Condition	. 440
DO COND Parameter	. 441
DO CTBRULE Parameter Format	
DO CTBRULE Parameter Example	
DO FORCEJOB Parameter Format	
DO FORCEJOB Parameter Example	. 446
DO FORCEJOB Confirmation Panel	. 446
<b>§Restart§</b> DO IFRERUN Parameter Format	
<b>§Restart§</b> DO IFRERUN Parameter Example	

DO MAIL Parameter Format	
DO MAIL Parameter Example	454
DO NOTOK Parameter Format	455
DO NOTOK Parameter Example	456
DO OK Parameter Format	
DO OK Parameter Example	
DO REMEDY Parameter Format	460
DO REMEDY Example	
DO RERUN Parameter Format	462
DO RERUN Parameter Example	
DO SET Parameter Format	465
DO SET Parameter Example	468
DO SHOUT Parameter Format	469
DO SHOUT Subparameter Example	. 474
DO STOPCYCL Parameter Format	475
DO STOPCYCL Parameter Example	. 476
DO SYSOUT Parameter Format	
Effect of Merging Multiple SYSOUT Statements	482
DO SYSOUT Parameter – Example 1	
DO SYSOUT Parameter – Example 2	
DOC Parameter Format	
DOC Parameter Example	487
DOCLIB Parameter Format	
DOCLIB Parameter Example	
DOCMEM Parameter Format	
DOCMEM Parameter Example	
DUE OUT Parameter Format	
DUE OUT Parameter Example	
GROUP Parameter Format	
GROUP Parameter Example	
GRP MAXWAIT Parameter Format	
GRP MAXWAIT Parameter Example	
IN Parameter Format	
Long IN Condition	
IN Parameter – Example 1	
IN Parameter – Example 2	507
IN Parameter – Example 3	
IN Parameter – Example 4	
IN Parameter – Example 5	
IN Parameter – Example 6	
INSTREAM JCL parameter format	
INTERVAL Parameter Format	
INTERVAL Parameter Example	
MAXRERUN Parameter Format	
MAXRERUN Parameter Example 2	
MAXWAIT Parameter Format	
MAXWAIT Parameter Example 1	
MAXWAIT Parameter Example 3	522
MEMLIB Parameter Format	
	220

MEMNAME Parameter Format	528
MEMNAME Parameter Example	
MINIMUM Parameter Format	
MINIMUM Parameter – Example 1	
MINIMUM Parameter – Example 2	
MONTHS Parameter Format	
MONTHS Parameter – Example 2	
NJE NODE Parameter Format	
ON Statement Format Example	
ON GROUP-END Parameter Format	
ON GROUP-END Parameter Example	
ON GROUP-END Confirmation Panel	
ON Parameter Format	
ON PGMST Parameter – Example 1	
ON PGMST Parameter – Example 2	
ON PGMST Parameter – Example 3	
OUT Parameter Format	
Long OUT Condition	
OUT Parameter Example 1 – First Job	
OUT Parameter Example 1 – Second Job	
OUT Parameter – Example 2	
OUT Parameter – Example 3	5/0
OVERLIB Parameter Format	
OVERLIB Parameter Example	
OWNER Parameter Format	
OWNER Parameter Example	
PDS Parameter Format	
PDS Parameter Example	
PIPE Parameter Format	
PIPE Parameter Example – Job CTLIVPWR	
PIPE Parameter Example – Job CTLIVPRD	
§Restart§ PREVENT-NCT2 Parameter Format	
<b>§Restart§</b> PREVENT-NCT2 Parameter Example	
PRIORITY Parameter Format	
RELATIONSHIP Parameter Format	
RELATIONSHIP Parameter Example	
RERUNMEM Parameter Format	
RERUNMEM Parameter Example	
RESOURCE Parameter Format	
RESOURCE Parameter – Example 1A	600
RESOURCE Parameter – Example 1B	601
RESOURCE Parameter – Example 1C	602
RETENTION: # OF DAYS TO KEEP Parameter Format	
RETENTION: # OF DAYS TO KEEP Parameter Example	604
RETENTION: # OF GENERATIONS TO KEEP Parameter Format	605
RETENTION: # OF GENERATIONS TO KEEP Parameter Example	606
RETRO Parameter Format	
RETRO Parameter – Example 1	608

SAC Parameter Format	
SAC Parameter Example	611
SCHEDULE TAG Parameter Format	612
SCHEDULE TAG Parameter – Example 1	614
SCHEDULE TAG Parameter – Example 2	615
SCHEDULE TAG ACTIVE Parameter Format	
SCHENV Parameter Format	619
SET VAR Parameter Format	621
SET VAR Parameter Example – 2A	625
SET VAR Parameter Example 2B	
SHOUT Parameter Format	
SHOUT Parameter Example 2	636
SHOUT and DO SHOUT Example	
STAT CAL Parameter Format	
STEP RANGE Parameter Format	641
STEP RANGE Parameter Example	
SYSOUT Parameter Format	
Merging SYSOUT and DO SYSOUT Statements	649
SYSOUT Parameter – Example 2	
SYSTEM ID Parameter Format	
TASKTYPE Parameter Format	654
TASKTYPE Parameter – Example 4	
TIME + DAYS Parameter Format	
FROM TIME Parameter Example	
TIME + DAYS Parameter Example	
TIME ZONE Parameter Format	
WDAYS Parameter Format	669
WDAYS Parameter Example 1	
WDAYS Parameter Example 2	
WDAYS Parameter Example 3	
WDAYS Parameter Example 4	
WDAYS Parameter Example 5	
WDAYS Parameter Example 6	
WDAYS Parameter Example 7	
WDAYS Parameter Example 8	
WDAYS Parameter Example 9	
WDAYS Parameter Example 10	
CMEM Rule Definition Screen	
DESCRIPTION Parameter Format	697
DESCRIPTION Parameter Example	
DO Parameter Format	
DO COND Parameter Format	
DO COND Parameter – Example 1	
DO COND Parameter – Example 2	
DO FORCEJOB Parameter Format	
DO FORCEJOB – Example 1	
DO FORCEJOB – Example 2	
DO RULE Parameter Format	
DO RUI F Evample	709

DO SHOUT Parameter Format	. 710
DO SHOUT Parameter Example	. 714
DO STOPJOB Parameter Format	. 715
DO STOPJOB Parameter Example	
GROUP Parameter Format	
GROUP Parameter Example	
MODE Parameter Format	
MODE Parameter Example	
ON Parameter Format	
ON DSNEVENT Parameter Format	
ON DSNEVENT Parameter Example	
ON JOBARRIV Parameter Format	
ON JOBARRIV Parameter Example	
ON JOBEND Parameter Format	
ON JOBEND Parameter Example	
ON STEP Parameter Format	
ON STEP Parameter Example	
OWNER Parameter Format	
OWNER Parameter Example	
RUNTSEC Parameter Format	
RUNTSEC Parameter Example	
THRESHOLD Parameter Format	
THRESHOLD Parameter Example	
Illustration 1A: How CONTROL-M Formerly Handled A New Tape	
Illustration 1B: Steps Formerly Performed by the User	
Illustration 2A: How CONTROL-M Now Handles A New Tape	
Illustration 2B: Single Step Now Performed by the User	
Parameter Prompting Facility Type 2: Definition Phase	
Parameter Prompting Facility Type 2: Fetch Phase	. 830
Parameter Prompting Facility Type 2: EXEC Phase	
The FETCH A PLAN Screen	
The EXEC / ORDER A PLAN Screen	
PPF2DEL Utility Screen	
CONTROL-M Simulation Exit Screen	
Sample Tape Pull List Report 1	. 857
Sample Tape Pull List Report 2	
Sample Tape Pull List Report 3	
Sample Tape Pull List Report 4	
JCL for USS Execution	
CONTROL-M Architecture for Unix-Oriented MVS Implementation	. 907
Architecture of SAP R/3	. 908
Communication with the R/3 Application Layer - DB/2 Database	
Communication with the R/3 Application - SAP/R3 Database	
The Edit Environment in The Job Scheduling Definition Screen	
Example - Inserting A DO Statement - Before	
Example - Inserting A DO Statement - After	
Example - Deleting A Block - Before	
Example - Deleting A Block - After	. 920
Example - Moving Statements - Before	. 921

Example - Moving Statements - After	922
Example - Copying Statements - Before	923
Example - Copying Statements - After	924
Example - Inserting A Line - Before	924
Example - Inserting A Line - After	925
The Edit Environment in The Rule Definition Screen	927
Example - Repeating A DO Block - Before	931
Example - Repeating A DO Block - After	931
Example - Automatic Restart - CONTROL-M Only	935



# **Tables**

List of INCONTROL Products
Job Scheduling Definition Sections
Runtime Criteria
Conditions and Resources
NJE Network Nodes
Event Types Handled by CMEM 52
KSL Report Types 55
IOA Core Files
Date Definition Types
Gregorian Date Notation
Supported Gregorian Dates
Julian Date Notation
Group Handling Criteria 69
Prerequisite Condition Statements
Runtime Scheduling Criteria
Prefixing Examples
Masking Examples
INCONTROL Shared IOA Functions and Facilities
CONTROL-M Functions and Facilities
IOA Primary Option Menu Options 88
IOA Transfer Control Commands
Basic IOA Screen Areas
Common PFKey Definitions
Additional Key Assignments
Scrolling Commands 95
Scrolling Amounts in the SCROLL Field
ISPF Commands that must be defined for PFKeys
PFKey Functions Within the IOA Editor Screen
IOA Editor Command Line Commands
IOA Editor Row Commands
Scheduling Definition Facility Screens
Scheduling Criteria 111
Options of the Table List Screen
Commands of the Job List Screen
Options of the Job List Screen
Parameters of the Job Scheduling Definition Screen
General Job Parameters
Basic Scheduling Parameters
Runtime Scheduling Parameters
Post-Processing Parameters

Parameters of the Group Entity Scheduling Definition Screen	. 140
Commands of the Job Scheduling Definition Screen	. 142
Save Documentation Window Fields	. 147
Commands for Exiting the Job Definition Screen	. 148
Options for Manually Ordering Jobs	. 151
Order and Force Confirmation Window Fields	. 153
Fields in the Window for Copying Jobs to Another Table	. 157
Color Change Options on Graphic Jobflow Window	
Job Scheduling Plan Window Fields	. 164
Job Scheduling Plan Screen Fields	. 166
Default Colors for Active Missions Screen	. 168
Predefined Display Types	. 169
Fields in the Default Display Type	. 171
Fields for Each Job Default	
Other Information in the STATUS Field	. 172
Fields in the All Fields Display Type	. 173
Commands of the Active Environment Screen	
Options of the Active Environment Screen	. 180
Job Statuses for the Active Environment screen	. 185
Group Statuses for the Active Environment Screen	. 189
Field of the Display Filters Window	. 191
Options of the Display Filters window	. 191
Fields of the Show Screen Filter Window	
Show Screen Filter Window Selection Criteria	. 193
Show Screen Filter Window - Closing Values	. 199
Fields of the Global View Screen	. 200
Fields of the View Graph Screen	. 202
Job Status Color	. 203
Fields of the View Graph Screen	
Job Graph Status Symbols	. 205
Fields of the Add Condition or Delete NOT-COND Confirmation Window	
Fields of the Job Scheduling Definition Zoom Screen	
Commands of the Zoom Screen	
Fields of the Confirm Rerun Window	. 223
<b>§Restart§</b> Fields of the Active Environment Rerun and/or Restart Confirmation	
Window	. 225
<b>§Restart§</b> Fields of the Flow Rerun Options Window	. 230
<b>Fields</b> of the Rerun Flow Job List Window	
Options of the Rerun and/or Restart Step List Window	
<b>§Restart§</b> Default Display Type Fields of Job Order Execution History Screen	. 235
<b>§Restart§</b> Fields in the Job Order Execution History Screen	. 236
<b>§Restart§</b> Commands of the Sysout Viewing Screen	
Statistics Screen Individual Execution Statistics	. 239
Statistics Screen Group Entity Execution Statistics	. 240
Fields of the Job Dependency Network Display Type N (Network)	
Parameter of the REFRESH Command	
Rule Definition Facility Screens	
Fields of the Entry Panel	. 253
Ontions of the Table List Screen	255

Fields of the Rule List Screen	
Commands of the Rule List Screen	
Options of the Rule List Screen	
CMEM Rule Definition Event Selection Parameter	259
CMEM Rule Definition General Parameters	260
CMEM Rule Definition Action Parameters	260
Commands of the Rule Definition Screen	262
Commands for Exiting the Rule Definition Screen	264
Options for Ordering Rule Tables	268
Fields in the Window for Copying Rules to Another Table	
IOA Variable Database Facility Screens	
Fields of the IOA Values of Database Screen	
Options of the Values of Database Screen	
Display Types of the Variable Zoom Screen	
Options of the Variable Zoom Screen	
Fields of the IOA Conditions/Resources Screen	
IOA Conditions/Resources Retrieval Criteria	
IOA Conditions/Resources ADD Command Formats	
Options of the IOA Conditions/Resources Screen	
IOA Conditions/Resources DELETE Confirmation Window Options	
COUNT Parameter Values	
Fields of the WHY Option Window	
Fields of the IOA Manual Conditions Screen	
Retrieval Criterion for IOA Manual Conditions Screen	
Options of the IOA Manual Conditions Screen	
Fields of the IOA Log Screen	
Commands of the IOA Log Screen	
IOA Log Screen Predefined Display Types	
Fields of the Display Filters Window	
Options of the Display Filters Window	
Fields of the IOA Log Show Screen Window	
	スリバ
IOA Log Show Screen Window Selection Criteria	
IOA Log Show Screen window - Closing Values	306
IOA Log Show Screen window - Closing Values	306 307
IOA Log Show Screen window - Closing Values	306 307 309
IOA Log Show Screen window - Closing Values	306 307 309 310
IOA Log Show Screen window - Closing Values IOA Calendar Facility Screens Fields of the IOA Calendar Facility Entry Panel Options of the Calendar List Screen Commands of the Year List Screen	306 307 309 310 312
IOA Log Show Screen window - Closing Values IOA Calendar Facility Screens Fields of the IOA Calendar Facility Entry Panel Options of the Calendar List Screen Commands of the Year List Screen Options of the Year List Screen	306 307 309 310 312 312
IOA Log Show Screen window - Closing Values IOA Calendar Facility Screens Fields of the IOA Calendar Facility Entry Panel Options of the Calendar List Screen Commands of the Year List Screen	306 307 309 310 312 312
IOA Log Show Screen window - Closing Values IOA Calendar Facility Screens Fields of the IOA Calendar Facility Entry Panel Options of the Calendar List Screen Commands of the Year List Screen Options of the Year List Screen Fields of the Calendar List Screen Copy Window Fields of the Calendar Definition Screen	306 307 309 310 312 312 314 316
IOA Log Show Screen window - Closing Values IOA Calendar Facility Screens Fields of the IOA Calendar Facility Entry Panel Options of the Calendar List Screen Commands of the Year List Screen Options of the Year List Screen Fields of the Calendar List Screen Copy Window	306 307 309 310 312 312 314 316
IOA Log Show Screen window - Closing Values IOA Calendar Facility Screens Fields of the IOA Calendar Facility Entry Panel Options of the Calendar List Screen Commands of the Year List Screen Options of the Year List Screen Fields of the Calendar List Screen Copy Window Fields of the Calendar Definition Screen Commands for Exiting the Calendar Definition Screen Prerequisite Condition Utility Screen Fields	306 307 309 310 312 314 316 321 325
IOA Log Show Screen window - Closing Values IOA Calendar Facility Screens Fields of the IOA Calendar Facility Entry Panel Options of the Calendar List Screen Commands of the Year List Screen Options of the Year List Screen Fields of the Calendar List Screen Copy Window Fields of the Calendar Definition Screen Commands for Exiting the Calendar Definition Screen Prerequisite Condition Utility Screen Fields	306 307 309 310 312 314 316 321 325
IOA Log Show Screen window - Closing Values IOA Calendar Facility Screens Fields of the IOA Calendar Facility Entry Panel Options of the Calendar List Screen Commands of the Year List Screen Options of the Year List Screen Fields of the Calendar List Screen Copy Window Fields of the Calendar Definition Screen Commands for Exiting the Calendar Definition Screen Prerequisite Condition Utility Screen Fields Parameters of the Job Request Utility Screen	306 307 309 310 312 314 316 321 325 327
IOA Log Show Screen window - Closing Values IOA Calendar Facility Screens Fields of the IOA Calendar Facility Entry Panel Options of the Calendar List Screen Commands of the Year List Screen Options of the Year List Screen Fields of the Calendar List Screen Copy Window Fields of the Calendar Definition Screen Commands for Exiting the Calendar Definition Screen Prerequisite Condition Utility Screen Fields	306 307 309 310 312 314 316 321 325 327 330
IOA Log Show Screen window - Closing Values IOA Calendar Facility Screens Fields of the IOA Calendar Facility Entry Panel Options of the Calendar List Screen Commands of the Year List Screen Options of the Year List Screen Fields of the Calendar List Screen Copy Window Fields of the Calendar Definition Screen Commands for Exiting the Calendar Definition Screen Prerequisite Condition Utility Screen Fields Parameters of the Job Request Utility Screen JCL Library Mode Parameters	306 307 309 310 312 314 316 321 325 327 330 331
IOA Log Show Screen window - Closing Values IOA Calendar Facility Screens Fields of the IOA Calendar Facility Entry Panel Options of the Calendar List Screen Commands of the Year List Screen Options of the Year List Screen Fields of the Calendar List Screen Copy Window Fields of the Calendar Definition Screen Commands for Exiting the Calendar Definition Screen Prerequisite Condition Utility Screen Fields Parameters of the Job Request Utility Screen JCL Library Mode Parameters Scheduling Library Mode Parameters	306 307 319 312 312 314 316 321 325 327 330 331 331

Define Parameters and Conditions Screen – Format	343
Define Parameters and Conditions Screen – Options	344
Fields of the Update Parameters and Set Conditions Screen	347
Fields of the Define Parameters in the Master Plan Screen	
Options of the Define Parameters in the Master Plan Screen	
Define Parameters in the Master Plan Screen - Exit Screen Commands	
Fetch Plan Screen OVERRIDE DAILY PLAN Values	
Format of the Update Parameter Values Screen	
Quick Schedule Definition Process	
Fields of the CONTROL-M Quick Schedule Definition Screen	
Prerequisite Condition Format Fields	
Formats for Prerequisite Conditions	
Fields that Affect Prerequisite Conditions Formats	
Fields in the Job List Screen	
Options of the Job List Screen	
General Parameters	
Category A, B, C, and D Parameters	
Runtime Scheduling Parameter Criteria	
Final Job Statuses	
Parameters Performed When the Job Ends OK	
Conditional Processing Statements	387
Return and Cyclic Post-Processing Parameters	
Group Entity Post-Processing Parameters	388
ADJUST CONDITIONS Parameter Values	
<b>§Restart§</b> AUTO-ARCHIVE Subparameter Format	
Optional CONFCAL Subparameters	
Optional CONFIRM Parameter Values	
Mandatory CONTROL Subparameters	
Optional ČONTROL Subparameter	
Optional CTB STEP Parameters	
DAYS Subparameters	420
Non-Periodic Scheduling Formats	
Periodic Scheduling Formats	
DEFINITION ACTIVE Subparameters	
DO Actions	
Relationship of DO Statements with Other Statements	435
DO COND Mandatory Subparameter Formats	437
DO CTBRULE Subparameters	
DO FORCEJOB Subparameter Formats	
<b>§Restart§</b> DO IFRERUN Subparameter Formats	448
DO MAIL Subparameter Formats	452
DO REMEDY Subparameter Formats	460
DO SET Subparameter	
DO SHOUT Subparameters	
DO SYSOUT Subparameters	
Varying Effect of SYSOUT Handling Operations	
GRP MAXWAIT Parameter Values	
IN Subparameters	
Date Reference Values – Example 6	

INTERVAL Subparameters	513
MAXWAIT Parameter Values	
MEMLIB Parameter Values for Non-Started Tasks	
MEMLIB Parameter Formats for Started Tasks	
MONTHS Parameter Values	533
ON Statement types	537
ON GROUP-END Values	
ON PGMST Parameter Subparameters	
ON PGMST Parameter CODES Values	551
ON PGMST Parameter Code Qualifiers	555
ON SYSOUT Values	559
OUT Mandatory Subparameters	562
OVERLIB Parameter: Algorithm for LIbraries Used when Option J (JCL) is	Specified
573	•
<b>§Restart§</b> PREVENT-NCT2 Subparameters	584
RELATIONSHIP Parameter Values	
RELATIONSHIP Parameter Scheduling Types	591
Mandatory RESOURCE Subparameters	596
Optional RESOURCE Subparameters	
RETENTION: # OF DAYS TO KEEP Parameter Values	
RETENTION: # OF GENERATIONS TO KEEP Values	605
RETRO Values	607
SAC Parameter Values	
SCHEDULE TAG ACTIVE Subparameters	
SHOUT Subparameters	
STEP RANGE Subparameters	
SYSOUT Subparameters	
TASKTYPE Parameter Values	
TASKTYPE Basic Type Values	
WDAYS Subparameters	
Non-Periodic Scheduling Formats	
Periodic Scheduling Formats	
Events handled by CMEM	
CMEM AutoEdit Variables	
CMEM Event Selection Parameters	
CMEM General Parameters	
CMEM Action Parameters	
DO Parameter Actions	699
DO COND Subparameters	700
DO FORCEJOB Subparameters	
DO RULE Subparameters	
DO SHOUT Subparameters	
DO SHOUT OPER Subparameter Examples	
MODE Parameter Values	
ON Parameter Statements	
DSNEVENT Subparameters	724
Valid STEPRC Code Qualifiers	
ON JOBARRIV Subparameters	
JOBEND Subparameters	~~.

ON STEP Subparameters	733
ON STEP Subparameter STEPRC Qualifiers	736
Valid RUNTSEC Values	
AutoEdit Control Statements	746
Non-Date AutoEdit System Variables	748
Date AutoEdit System Variables	751
4 Character Year Date AutoEdit System Variables	752
Special AutoEdit System variable resolved after a group is ordered but before jo	
submission	753
Special AutoEdit System Variables Resolved after Job End	753
Special AutoEdit System Variable Resolved after Job Submission	
IOA Global Variable Database Structure Levels	761
Chart for Determining Priorities of Value Assignment Sources	770
%%RESOLVE Statements	778
AutoEdit Operators	781
Date Variables	797
Alternative Job Ordering Methods	813
Parameter Prompting Facility Type 2: Use of CLISTS	832
The FETCH A PLAN Screen: Parameters	833
The EXEC / ORDER A PLAN Screen: Parameters	834
Files Used as Input during Simulation	841
Files Produced as Output of Simulation	841
Parameters Passed to the Utility by DASIMPRM	
Online Simulation Environment File Allocations	
Overrides To Be Specified on IOALDNRS	850
Overrides To Be Specified on ADDMNCND	
Override To Be Specified for Simulation Step	851
CTMTAPUL Subparameters	
DD statements used by CTMTAPUL	
Order or Force Return Codes	
Files Accessed during the Order or Force Process	
Create and/or Order or Force New Tables Return Codes	
Files Accessed during the Create, Order or Force Process	
AJF Action Return Codes	871
Files Accessed during the AJF Action Process	872
Search Action Return Codes	
Files Accessed during the AJF Action Process	
Selection Criteria Parameters	
Selection Criteria Parameter Attributes	
Files Accessed during the AJF Action Process	
Contents of Registers on Input to CTMAPI	
Fixed Part Values	
Status Extension Fields	
Statuses Returnable under the Status Function	
Status Return Codes	
Files Accessed during the AJF Action Process	
Order Fields	
Order Return Codes	
ΔIF Δction Parameters	296

AJF Action BAPIAACT Field Values	896
CTMAPI Action Return Codes	897
Global Variable Fields	898
Global Variable Return Codes	899
CTMAPI Quantitative Resource Fields	899
CTMAPI Quantitative Resource Return Codes	900
CTMAPI Quantitative Resource File Allocation	900
BLT Action Return Codes	901
CTMAPI Quantitative Resource File Allocation	902
CTMAPI Reply Mechanism Trigger Pointers	
Subjects of Line Editing Commands	914
Line Editing Commands - Delete Commands	915
Line Editing Commands - Copy Commands	915
Line Editing Commands - Move Commands	916
Line Editing Commands - Repeat Commands	
Line Editing Commands - Insert Command	916
Line Editing Commands - Location Commands	916
Line Editing Commands - Delete Commands	928
Line Editing Commands - Copy Commands	928
Line Editing Commands - Move Commands	929
Line Editing Commands - Repeat Commands	929
Line Editing Commands - Insert Command	
Line Editing Commands - Location Commands	930



# **About This Guide**

CONTROL-M is a component member of the INCONTROLTM by BMC Software family of products. The *CONTROL-M for z/OS User Guide* is the main publication that describes the components and usage of CONTROL-M software.

This guide is designed for use by everyone who defines job schedules or who uses CONTROL-M to actively control jobs in the production environment.

This guide provides detailed information on all CONTROL-M functions and facilities. It contains the following chapters:

## **Chapter 1-Introduction**

CONTROL-M introduction and overview. This chapter briefly describes the main components of CONTROL-M from a functional perspective, and introduces the user to CONTROL-M facilities and features, concepts and logic. INCONTROL and IOA components and concepts are also described.

It is highly recommended that all users read this chapter before reading other chapters in the guide.

#### **Chapter 2–Online Facilities**

Guide to using CONTROL-M and IOA online facilities. CONTROL-M and IOA screens are illustrated and discussed in logical sequence.

#### **Chapter 3–Job Production Parameters**

Detailed description, accompanied by examples, of the parameters and statements in the CONTROL-M Job Scheduling Definition screen.

#### Chapter 4–CONTROL-M Event Manager (CMEM)

Overview of CONTROL-M Event Manager logic and a detailed description of the parameters and statements in CMEM rule definitions. This facility enables CONTROL-M to respond to external events (that is, events in the MVS environment that occur outside of CONTROL-M's direct control).

#### Chapter 5-JCL and AutoEdit Facility

Guide to the CONTROL-M AutoEdit facility, and its application to JCL. Usage of AutoEdit terms in the JCL can eliminate the need for manual changes to the JCL prior to job submission.

#### **Chapter 6–Selected Implementation Issues**

Provides concepts, hints, and procedures for successful implementation and maintenance of CONTROL-M.

### **Chapter 7–Simulation and Forecasting Facility**

Guide to simulating the effects of operations and procedures in your production environment and forecasting the potential impact of proposed changes.

### Appendix A-Editing Job Scheduling Definitions in the Edit Environment

Usage of the IOA Edit environment for editing job scheduling definitions.

### Appendix B-Editing CMEM Rule Definitions in the Edit Environment

Usage of the IOA Edit environment for editing CMEM rule definitions.

## Appendix C-MVS Job Restart Without CONTROL-M/Restart

Instructions for using CONTROL-M to perform an MVS restart (for sites that do not have CONTROL-M/Restart installed).

Index

# **Conventions Used in This Guide**

Notational conventions that may be used in this guide are explained below.

#### **Standard Keyboard Keys**

Keys that appear on the standard keyboard are identified in boldface, for example, **Enter**, **Shift**, **Ctrl+S** (a key combination), or **Ctrl S** (a key sequence).

#### — WARNING -



The commands, instructions, procedures, and syntax illustrated in this guide presume that the keyboards at your site are mapped in accordance with the EBCDIC character set. Certain special characters are referred to in this documentation, and you must ensure that your keyboard enables you to generate accurate EBCDIC hex codes. This is particularly true on keyboards that have been adapted to show local or national symbols. You should verify that

\$ is mapped to x'5B' # is mapped to x'7B' @ is mapped to x'7C'

If you have any questions about whether your keyboard is properly mapped, contact your system administrator.

#### **Preconfigured PFKeys**

Many commands are preconfigured to specific keys or key combinations. This is particularly true with regard to numbered PF keys, or pairs of numbered PFKeys. For example, the END command is preconfigured to, and indicated as, **PF03/PF15**. To execute the END command, press either the **PF03** key or the **PF15** key.

Instructions to enter commands may include

- only the name of the command, such as, enter the END command
- only the PF keys, such as, press PF03/PF15
- or both, such as, press PF03/PF15, or enter the END command

### **Command Lines and Option Fields**

Most screens contain a command line, which is primarily used to identify a single field where commands, or options, or both, are to be entered. These fields are usually designated COMMAND, but they are occasionally identified as COMMAND/OPT or COMMAND/OPTION.

Option field headings appear in many screens. These headings sometimes appear in the screen examples as OPTION, or OPT, or O.

Names of Commands, Fields, Files, Functions, Jobs, Libraries, Members, Missions, Options, Parameters, Reports, Subparameters, and Users

The names of commands, fields, functions, jobs, libraries, members, missions, options, parameters, reports, subparameters, users, and most files, are shown in standard UPPERCASE font.

#### **User Entries**

In situations where you are instructed to enter characters using the keyboard, the specific characters to be entered are shown in this **UPPERCASE BOLD** text, for example, type **EXITNAME**.

#### **Syntax Statements**

In syntax, the following additional conventions apply:

■ A vertical bar ( | ) separating items indicates that you must choose one item. In the following example, you would choose *a*, *b*, or *c*:

```
a | b | c
```

- An ellipsis (...) indicates that you can repeat the preceding item or items as many times as necessary.
- Square brackets ([]) around an item indicate that the item is optional. If square brackets ([]) are around a group of items, this indicates that the item is optional, and you may choose to implement any single item in the group. Square brackets can open ([) and close (]) on the same line of text, or may begin on one line of text and end, with the choices being stacked, one or more lines later.
- Braces ({ }) around a group of items indicates that the item is mandatory, and you must choose to implement a single item in the group. Braces can open ({) and close (}) on the same line of text, or may begin on one line of text and end, with the choices being stacked, one or more lines later.

#### **Screen Characters**

All syntax, operating system terms, and literal examples are presented in this typeface. This includes JCL calls, code examples, control statements, and system messages. Examples of this are:

calls, such as

#### CALL 'CBLTDLI'

code examples, such as

```
FOR TABLE owner.name USE option, . . . ;
```

control statements, such as

#### //PRDSYSIN DD * USERLOAD PRD(2) PRINT

■ system messages, both stand-alone, such as You are not logged on to database database_name, and those embedded in text, such as the message You are not logged on to database database_name, are displayed on the screen.

#### **Variables**

Variables are identified with *italic* text. Examples of this are:

- In syntax or message text, such as Specify database database_name
- In regular text, such as replace database *database_name1* with database *database_name2* for the current session
- In a version number, such as EXTENDED BUFFER MANAGER for IMS 4.1.xx

### **Special elements**

This book includes special elements called *notes* and *warnings*:

#### _ NOTE -



Notes provide additional information about the current subject.



#### WARNING -

Warnings alert you to situations that can cause problems, such as loss of data, if you do not follow instructions carefully.

# Information New to This Version

Where substantive additions and modifications to the content of this guide occur, revision bars have been inserted in the margin.

Additional information that is new to this version is described in Appendix C of the *INCONTROL for z/OS Upgrade Guide*.

# **Information Relating to CONTROL-M/Restart**Users

Certain information presented in the CONTROL-M for z/OS User Guide is relevant only to CONTROL-M users who have CONTROL-M/Restart installed at their site.

#### NOTE -



CONTROL-M/Restart was called CONTROL-R in earlier versions.

CONTROL-M/Restart information is identified in this guide by the **§Restart§** symbol, which is shown at the beginning of the CONTROL-M/Restart information. This symbol is shown using the following guidelines:

■ If an entire topic level is dedicated to CONTROL-M/Restart material, the heading of that topic begins with the **§Restart§** symbol. Similarly, if there are lower level topics within that level that are also dedicated to CONTROL-M/Restart material, the headings of those lower level topics will also begin with the **§Restart§** symbol.

This provision also applies to CONTROL-M/Restart paragraphs, each of which will begin with the **\$Restart\$** symbol, or, on occasion, to single sentences, or even phrases or words, if they exclusively pertain to CONTROL-M/Restart material.

- The same **§Restart§** symbol is placed at the conclusion of each unbroken block of text material that contains CONTROL-M/Restart material, regardless of whether the material spans more than one heading level, paragraph, or sentence. For example, if a first level CONTROL-M/Restart topic includes second and/or third and/or fourth and/or fifth level topic headings, with no intervening material that is not related to CONTROL-M/Restart, the **§Restart§** symbol will be placed at the end of the text in the lowest level sentence of unbroken CONTROL-M/Restart material.
- If a figure or table is used exclusively to identify or explain CONTROL-M/Restart material, the following statement will appear immediately preceding the figure title or the table title:

**§Restart§** The following (figure)(table) is for users who have CONTROL-M/Restart installed at their site.

If CONTROL-M/Restart material is included only in part of a figure or table otherwise used to illustrate standard CONTROL-M material, the §Restart§ symbol will be used within the figure or table to identify the information relevant only to CONTROL-M/Restart users. If CONTROL-M/Restart is not installed at your site, you can skip any material in this guide that is identified with the **\$Restart\$** symbol.

# **Related Publications**

### CONTROL-M for z/OS Getting Started Guide

Explanation of CONTROL-M facilities. Online, step-by-step instructions are provided.

#### INCONTROL for z/OS Administrator Guide

Information for system administrators about customizing and maintaining INCONTROL products.

#### INCONTROL for z/OS Installation Guide

A step-by-step guide to installing INCONTROL products using the INCONTROL TM Installation and Customization Engine (ICE) application.

### **INCONTROL** for z/OS Messages Manual

A comprehensive listing and explanation of all IOA and INCONTROL messages and codes.

### **INCONTROL** for z/OS Security Guide

A step-by-step guide to implementing security in INCONTROL products using the ICE application.

### **INCONTROL** for z/OS Utilities Guide

Describes utilities designed to perform specific administrative tasks that are available to INCONTROL products.

Chapter





# **Introduction to CONTROL-M**

This chapter includes the following topics:

INCONTROL Products and IOA	42
IOA	42
INCONTROL	42
Functional Approach	43
Main Components	
Expanded CONTROL-M Functionality	
CONTROL-M Support for MAINVIEW Batch Optimizer	
Online User Interface to CONTROL-M	
CONTROL-M Concepts	61
IOA Core and CONTROL-M Repository	
Date Definition Concepts	
Date Standards and Date Field Formats	65
Job Ordering and Job Forcing	66
Rerun and Restart	67
Order ID	68
SYSDATA	68
Handling of Job Groups	68
Prerequisite Conditions	
Quantitative and Control Resources	73
Job Priority	74
Automatic Job Flow Adjustment	74
<del>*</del>	

# **INCONTROL Products and IOA**

The CONTROL-M Automated Production Control and Scheduling System is a component member of the INCONTROL family of products, a fully integrated suite designed to automate, manage and streamline operations on the OS/390 or z/OS mainframes. The INCONTROL family also includes client and server products that facilitate the automation of other platforms.

### **IOA**

The Integrated Operations Architecture (IOA) is at the heart of the INCONTROL family of products. IOA has a common core of shared code as the foundation of its architecture design. INCONTROL's IOA environment has several inherent design advantages, including a common user interface and a shared data repository. A key feature of the IOA environment is its integrated application design, which includes:

- Integrated User Notification
- Management by Exception
- **■** Integrated Scheduling
- Interdependency and Interrelationship Handling
- **■** Common Help Facility
- Integrated Management Reporting
- Common Method for Sharing Information
- Unified Installation and Maintenance
- Unified Security Implementation
- Open Interface Design

### **INCONTROL**

The INCONTROL family of products includes:

Table 1 List of INCONTROL Products (part 1 of 2)

Product	Description
CONTROL-M	Automated Production Control and Scheduling System Manages and automates the setup, scheduling and execution of jobs in the data center.
CONTROL-M/Restart	Restart Management System Automates the activities that must be performed when restarting failed jobs, including the scratching and uncataloging of data sets created by failed jobs.

Table 1 List of INCONTROL Products (part 2 of 2)

Product	Description
CONTROL-M/Tape	Removable Media Management System Increases utilization of removable media and controls retention periods. Prevents misuse of media, and provides tape library and vault control.
CONTROL-M/Analyzer	Automated Information Integrity System Performs in-stream validation, accuracy, and reasonability checks on information used by data center production tasks (for example, reports, databases).
CONTROL-D	Output Management System Automatically schedules and controls every aspect of report processing and distribution, including report decollating, bundling, printing, online viewing, and archiving.
CONTROL-V	Quick Access Archive Viewing System Provides online access to archived reports and documents by indexed data retrieval.
CONTROL-D/ Page On Demand	Report Retrieval and Display System Enables end users to retrieve and view pages of reports that reside on mainframe storage in real time. Indexed reports can be retrieved by index name and value. AFP and XEROX reports can also be retrieved and displayed using CONTROL-D/WebAccess Server or CONTROL-D/Page On Demand API.
CONTROL-D/ Image	Image Output Management System Enables output from commercial imaging equipment to be imported into either CONTROL-D or CONTROL-V for decollation, distribution and viewing, and into CONTROL-V for archiving and indexed retrieval.
CONTROL-O	Console Automation System and Desired State Monitoring System Monitors and automatically responds to messages, commands, and data set events, as well as various other system events.
	The CONTROL-O/COSMOS feature allows for status monitoring while maintaining all critical system objects in a desired and ideal status.

# **Functional Approach**

CONTROL-M automates job processing in your data center.

- It performs virtually all the job handling tasks of computer operators.
- It provides an interface that enables the user to intervene in the process of production management.

■ It provides continual data and status information regarding job processing.

CONTROL-M contains many facilities and components. Working together, they automate the data center. This chapter introduces the CONTROL-M facilities and components from a functional perspective, beginning with the major components that comprise the heart of CONTROL-M and progressing to the more minor components that enhance the functionality of CONTROL-M.

# **Main Components**

The following components are essential to CONTROL-M:

- Job scheduling definitions
- Active Jobs file
- **■** CONTROL-M monitor

# **Job Scheduling Definitions**

A job scheduling definition specifies criteria that identify decisions to be made, and actions to be taken, regarding the handling of a particular job. Each job scheduling definition contains the following sections:

lable 2	Job S	cheduling	Definition	Sections

Section	Description
General Parameters	General information about the job (for example, identifies the library and member in which the JCL is stored).
Basic Scheduling Parameters	Criteria according to which CONTROL-M schedules the job.
Runtime Scheduling Parameters	Runtime requirements that must be satisfied before CONTROL-M submits the job.
Post-processing Parameters	Actions CONTROL-M performs after the job ends, depending upon the outcome of job execution. For example, CONTROL-M performs one set of actions if the job ends OK, but another set of actions if an abend occurs.

Job scheduling definitions only need be defined once for each job in the production environment. The mechanism used to define job scheduling definitions is discussed in Chapter 2, "Online Facilities." Once defined, a job scheduling definition is saved. It can be modified later if required, and the changes saved.

Job scheduling definitions are stored in members in partitioned data sets (libraries), as follows:

- Job scheduling definitions for related applications are generally placed in a single member, called a scheduling table.
- Multiple scheduling tables are stored in partitioned data sets, called scheduling libraries.
- Multiple scheduling libraries can be defined.

### **Active Jobs File**

As mentioned above, each job scheduling definition contains criteria that determine whether the job must be scheduled on a given day. If based on these criteria a job must be scheduled, a copy of its job scheduling definition is placed in a file called the Active Jobs file. The mechanism by which job scheduling definitions are placed in the Active Jobs file is discussed in "Job Ordering and Job Forcing" on page 66.

Only jobs in the Active Jobs file are candidates for submission by the CONTROL-M monitor.

### **CONTROL-M Monitor**

The CONTROL-M monitor handles and controls job processing:

- It checks the runtime requirements specified in each job scheduling definition in the Active Jobs file, monitors available resources and conditions in the environment, and if it determines that the conditions and resources required by a job are available, it allocates the resources and submits the job.
- It monitors the execution of the job.
- It implements post-processing decisions based on instructions in the job scheduling definition and the results of the job execution.

The CONTROL-M monitor operates continually. It evaluates the production environment and implements decisions.

# **Expanded CONTROL-M Functionality**

This section describes facilities, features and capabilities of CONTROL-M which supplement the main components of the program.

# **Automating Job Scheduling: New Day Processing**

One of the main purposes of CONTROL-M is to automate job scheduling.

We have already explained that basic scheduling criteria for each job are defined in its job scheduling definition, and that a copy of the job scheduling definition is placed in the Active Jobs file when the basic scheduling criteria are satisfied.

The mechanism used to place job scheduling definitions automatically in the Active Jobs file is called New Day processing.

At a set time each day (defined during installation as the start of day at the site), CONTROL-M performs New Day processing, during which:

- CONTROL-M performs a number of maintenance and cleanup functions that the operator would otherwise have to perform manually.
- Job scheduling definitions are selected from the scheduling tables (based on their basic scheduling criteria) and are placed in the Active Jobs file. These jobs can then be submitted and tracked by the CONTROL-M monitor.

The implementation of automated job scheduling and New Day processing, and the components of New Day processing, are discussed in detail in the *INCONTROL* for *z/OS Administrator Guide*.

## Automatic JCL Update: JCL and AutoEdit Facility

In the production environment, JCL must often be manually modified prior to submission of a job, as in the following cases:

- changing a parameter or a date card
- supplying tape numbers in JCL procedures
- eliminating steps under different run conditions, for example, when end of month processing differs from normal daily run

Manual modification of the JCL is inconvenient at best, and it can be error-prone and lead to serious problems. The JCL and AutoEdit facility offers an automated alternative to manual JCL update.

The JCL and AutoEdit facility permits AutoEdit terms, such as AutoEdit variables, functions, and control statements, to be specified in the JCL in place of values that change from job submission to job submission. AutoEdit terms are prefixed by %%, which distinguishes them from non-AutoEdit terms. For example, the term %%ODAY is recognized as an AutoEdit term.

The values of user-defined variables that have been defined as Sysplex-wide, using the XAE facility, remain both in memory and in a Coupling facility. These values can be used for additional triggering of the same job or other CONTROL-M jobs, in the same computer or in different computers of the same Sysplex.

At time of job submission, AutoEdit terms in the JCL are resolved to their actual values.

The inclusion of AutoEdit terms into the job stream and job scheduling definitions can eliminate the need to change JCL once it is defined. AutoEdit usage can be further simplified and enhanced through the Parameter Prompting facility, which is described in "M4: Parameter Prompting Facilities" on page 337 and "Parameter Prompting Facilities" on page 823.

As of version 6.1.00, CONTROL-M/eTrigger can be used as an alternative to the Parameter Prompting Facility. AutoEdit parameter values can be passed together with the job scheduling definition when using the CONTROL-M/Enterprise Manager to order an unscheduled job. If this is done, these AutoEdit parameter values are substituted for those already in the job scheduling definition prior to submission.

For more information on CONTROL-M/eTrigger, see the *CONTROL-M/eTrigger Administrator Guide*.

The JCL and the AutoEdit facility is described in detail in Chapter 5, "JCL and AutoEdit Facility."

### **Automated Job Submission**

Once a job has been placed in the Active Jobs file, the CONTROL-M monitor does not submit the job unless all its runtime scheduling criteria, as defined in the job scheduling definition, are satisfied. Several types of runtime criteria can be defined.

### **Examples**

**Table 3** Runtime Criteria

Criteria	Description
Time	Submission must occur during a defined time range.
Priority	Jobs can be assigned internal priorities, so that if two jobs are ready for submission at the same time, the higher-priority job is submitted first.
Due Out	If two jobs with the same priority are ready for submission, the job with the earlier due out time is submitted first.

# **Monitoring of Resources**

Three types of runtime criteria require CONTROL-M to monitor the existence of conditions and the availability of resources system-wide. These conditions and resources are mentioned briefly below and are discussed in greater detail in "CONTROL-M Concepts" on page 61:

Table 4 Conditions and Resources

Condition or Resource	Description
Quantitative resources	Quantity of a resource required by the job. For example, a job may require two tape drives.
Control resources	Mode (exclusive or shared) in which a resource is required. For example, a backup job may require exclusive access to a specified data set.
Prerequisite conditions	User-defined conditions that must exist before a job is submitted. A major use of prerequisite conditions is to establish job dependencies.

The condition and resource requirements of a job are defined in the job scheduling definition.

Prerequisite conditions are tracked by the IOA Conditions file. Existing and available Quantitative resources and Control resources are tracked by the CONTROL-M Resources file. Prior to version 6.0.00, conditions and resources were stored in a single file, the Conditions/Resources file.

When the prerequisite conditions and resources required by a job are available, the job can be submitted by the monitor, if all other runtime scheduling criteria are satisfied.

# Immediate Detection and Notification of Problems: Shout Facility

When a problem or an unexpected situation or delay occurs, CONTROL-M can notify the appropriate personnel. These situations and problems are detected by analysis of a job sysout.

Notification is issued by the Shout facility, which can send messages to a variety of destinations including the operator console, a TSO user, and the IOA Log file.

CONTROL-M can also be instructed to issue a SHOUT message in the event an exception occurs at time of job submission and/or during job execution, such as when a job completes before, or later than, its anticipated completion time.

### **History Jobs File**

During New Day processing, jobs that have ended OK or whose retention period has expired according to job scheduling definition parameters are deleted from the Active Jobs file.

These jobs can be placed in the History Jobs file during New Day processing. This is an optional feature that can be activated by the INCONTROL administrator. Activation of this feature is described under the HIST parameter in the *INCONTROL* for *z/OS Administrator Guide*.

Jobs in the History Jobs file can by request be restored to the Active Jobs file, for subsequent restart.

Jobs remain in the History Jobs file until they are deleted according to criteria defined in the job scheduling definition.

The contents of the History Jobs file can be viewed from the History Environment screen, which is described in Chapter 2, "Online Facilities."

# **Journaling and Restoration Capability**

The CONTROL-M Journal file collects data about changes occurring in the CONTROL-M Active Jobs file, the IOA Conditions file and the CONTROL-M Resources file during the CONTROL-M working day. This permits forward recovery of the CONTROL-M environment to any time of the day you may choose.

The Journal file is initialized each day during New Day processing. From that point on, for the rest of the working day, the CONTROL-M monitor records in the Journal file all job processing activities that impact the CONTROL-M Active Jobs file, and all prerequisite condition additions to and deletions from the IOA Conditions file and the CONTROL-M Resources file.

If the CONTROL-M Active Jobs file, and optionally the IOA Conditions file and the CONTROL-M Resources file, need to be restored, for example, following a system crash, the CTMRSTR utility can be run to restore the files. The utility uses data from the Journal file to restore the files to the status they had at any specific time after the last run of the New Day procedure.

The CONTROL-M Journal file is initialized each day during New Day processing. Therefore, the time at which the New Day procedure initialized the Journal file is the earliest time to which the CONTROL-M Active Jobs file, the CONTROL-M Resources file, or the IOA Conditions file can be restored.

Journaling and Restoration is an optional feature that can be activated by the INCONTROL administrator. It is described in the *INCONTROL for z/OS Administrator Guide*. Activation of this feature is described under the JRNL parameter in the *INCONTROL for z/OS Installation Guide*.

### **IOA Log Facility**

Messages issued by CONTROL-M are written to the IOA Log file. The IOA Log file is a repository for messages issued by all INCONTROL products. Through the IOA Log facility, the user can examine messages issued by CONTROL-M during the processing of a job.

### **Automated Job Post-Processing**

Once the job has executed, the CONTROL-M monitor implements the post-processing instructions defined in the job scheduling definition. Post-processing instructions can be defined for virtually any situation, such as job ended successfully, job abended, a particular condition code occurred in a particular step, and so on.

As part of post-processing, CONTROL-M can do the following:

 add a prerequisite condition to, or delete a prerequisite condition from, the IOA Conditions file

This can trigger or prevent the submission of a job in the Active Jobs file.

- force the placement of a job scheduling definition into the Active Jobs file, regardless of the basic scheduling criteria of the job
- set AutoEdit variables
- send (shout) a specified message to a specified location through the SHOUT facility or by electronic mail
- send a message by mail to the recipient identified by the mail name prefix
- change the final status of a job to OK or NOTOK
- handle the job SYSOUT

This includes changing its class, deleting it, rerouting it to another node, releasing it for printing, or copying it to another location.

- if CONTROL-M/Analyzer is active, invoke a CONTROL-M/Analyzer rule
- rerun a job

- perform an MVS job restart; for more information, see "OUT: Post–Processing Parameter" on page 562
- **§Restart§** if CONTROL-M/Restart is active, perform a CONTROL-M/Restart job restart
- **§Restart§** if CONTROL-M/Restart is active, automatically archive certain portions of the job output
- stop recycling of cyclic jobs

### **Utilities**

Utilities provided with CONTROL-M are used to perform a variety of management functions and generate reports that assist in the efficient use of CONTROL-M. Batch utilities are described in the *INCONTROL* for *z/OS Utilities Guide*. Online utilities are described in Chapter 2, "Online Facilities."

# **Handling Jobs in the NJE Network**

The CONTROL-M monitor handles the control of complex distributed production environments where jobs may be routed for execution to different nodes of the NJE network according to the business needs of the enterprise.

CONTROL-M differentiates between host and remote nodes in the NJE network as follows:

Table 5 NJE Network Nodes

Node	Description
	NJE network node under which the CONTROL-M monitor is active and the NJE job is submitted to MVS/JES by the monitor.
Remote node	NJE network node to which a job was sent from the host node.

An NJE job is a job submitted by the CONTROL-M monitor for execution on a remote node. CONTROL-M can detect the status of jobs running on a remote node so that once these jobs finish executing, CONTROL-M can assign a status to them.

### Handling External Events: CMEM Facility

External events are events in the system that occur outside the control of the CONTROL-M monitor, such as the submission of a job. The CONTROL-M Event Manager (CMEM) facility enables CONTROL-M to respond to and handle such events.

Through rules defined online through the CMEM Rule Definition facility, which is described in Chapter 2, "Online Facilities," the user specifies actions CONTROL-M must perform in response to external events.

The following types of events are handled by the CMEM facility:

Table 6 Event Types Handled by CMEM

Event	Description
Job Arrival	Arrival of a job on the JES spool, from any source.
Job End	Completion of a job, regardless of its source.
Dataset Event	Either the setting of a data set disposition at deallocation time or the occurrence of a NOT CATLGD 2 event.
Step	Termination of a procedure (and optionally, a program) step.

The following actions can be performed by the CMEM facility:

■ force one or more CONTROL-M jobs

For more information, see "Job Ordering and Job Forcing" on page 66.

- add prerequisite conditions to, or delete prerequisite conditions from, the IOA Conditions file and the CONTROL-M Resources file
- stop the job in which the event occurs
- invoke a CONTROL-O rule, if CONTROL-O is active at the site
- send a message to a specified location using the CONTROL-O SHOUT facility, if CONTROL-O is active at the site
- bring under the control of the CONTROL-M monitor a job submitted outside the control of the CONTROL-M monitor, such as a job submitted by a TSO user

Such a job is called an On Spool job, and the control of On Spool jobs is one of the most important functions of CMEM.

The CMEM facility, and On Spool jobs, are described in Chapter 4, "CONTROL-M Event Manager (CMEM)."

# **Using Calendars to Schedule Jobs: IOA Calendar Facility**

Specification of scheduling criteria for jobs can be simplified by using calendars. A calendar is a defined schedule that can be applied to jobs, such as Mondays through Fridays in each week in each month.

Calendars are defined in the Calendar facility. Each calendar is assigned a unique name that can be specified in job scheduling definitions. A particular calendar (that is, schedule) need only be defined once.

Specifying the name of a calendar in job scheduling definitions causes that calendar to be used to schedule those jobs.

Two types of calendars can be defined:

- regular
- periodic

### **Regular Calendars**

Regular calendars consist of scheduling dates or days of the week that can be defined according to monthly patterns.

### For example

- WEEKDAYS schedules jobs each Monday through Friday in each month.
- WEEKENDS schedules jobs on every Saturday and Sunday in each month.
- QUARTERLY schedules jobs on the last date in each quarter: March 31, June 30, September 30, December 31.

Regular calendars are especially useful when many jobs have the same schedule. Defining the schedule once in a calendar, and entering the calendar name in the job scheduling definition of the jobs with that schedule, makes it unnecessary to individually define that schedule in each job scheduling definition.

#### **Periodic Calendars**

Periodic calendars are especially useful when scheduling dates do not easily conform to fixed date or day of the week or month patterns.

#### For example

■ PAYCAL – Calendar used for jobs that are scheduled every other Wednesday (such as payroll jobs). Scheduling occurs on the first, third, and (if there is one) fifth Wednesday of some months. Scheduling may occur on the second and fourth Wednesday of other months.

The IOA Calendar facility is described in Chapter 2, "Online Facilities."

### **Accumulating Statistics: Statistics Facility**

As part of the post-processing for each job, CONTROL-M determines the elapsed run time of the job. All accumulated information regarding job execution, including the elapsed run time, is written to the IOA Log file.

Periodically, a statistics utility may be used to scan and analyze the IOA Log file. This utility gathers information about the start time of each job, its elapsed run time, CPU utilization time, and so on. The utility places this information in the Statistics file, where averages of these values can be maintained for each job.

Statistics facility averages may be used for several purposes:

- to determine if the execution time of a job falls outside of a statistically normal range of time, which would indicate an execution delay or problem)
- to calculate DUE-IN time for use by the Deadline Scheduling facility, which is discussed under "Automatic Job Flow Adjustment" on page 74
- to determine when a shout message must be issued based on the elapsed time of the job
- to simulate job executions and forecast the impact of changes to the system (described briefly below)
- to determine if a job can complete execution before the CONTROL-M planned shutdown time (QUIESCE command)

# Simulating Job Execution and Forecasting Resource Usage: Simulation and Forecasting Facility

Using statistics accumulated by the Statistics facility, the Simulation and Forecasting facility simulates the actions of the CONTROL-M monitor under the conditions specified in simulation parameters.

The Simulation and Forecasting facility enables you to forecast anticipated job load for a specified time in the future, and to forecast the effects of possible changes to the system, such as the impact of:

- removing four tape drives
- increasing CPU power by 30%
- changing the time at which certain jobs are executed

The Simulation and Forecasting facility can improve the efficiency of your site. It can help with resource and configuration decisions, and it can help with the planning of workload scheduling to achieve maximum utilization of resources.

### **Automatic Tape Adjustment**

The Automatic Tape Adjustment facility collects and analyzes statistics regarding tape drive usage, and automatically allocates the appropriate number of tape drives at job order time. This facility, which can be implemented by your INCONTROL administrator, overrides any tape drive Quantitative resource value specified in the job scheduling definition. For more information, see "Statistics Screen" on page 238 and "RESOURCE: Runtime Scheduling Parameter" on page 596.

# **Reporting Facility**

CONTROL-M supports a comprehensive reporting facility, which can produce the following types of reports:

Table 7 KSL Report Types

Reports	Description
Keystroke Language Reports	These are reports generated with the Keystroke Language (KSL). KSL is a general purpose reporting language, based on the Online facility, capable of producing numerous reports from the database, and is described in the <i>KeyStroke Language (KSL) User Guide.</i>
Special Purpose Reports	These reports include the Job Flow reports that are generally used to track the dependencies between jobs, and the Job Plan reports that are used to anticipate which jobs are scheduled each day.

Sample reports are provided in the IOA SAMPLE library. The Reporting facility is described in the *KeyStroke Language (KSL) User Guide*.

### Minus-One Support

Minus-One support is provided as part of CONTROL-M and enhances Parallel Sysplex support (CTMPLEX). With Minus-One support, CONTROL-M users that implement several CONTROL-M monitors in a Sysplex environment can run several installations of CONTROL-M with different maintenance releases or different versions, in parallel. This enables CONTROL-M users to implement installation and upgrade procedures without having to shut down their entire complex.

Minus-One support is available even at sites that are not operating in a Sysplex environment.

#### WARNING



When upgrading a specific CONTROL-M instance to a new release, you must not utilize features of the new release until all other components (members of the Sysplex, application servers, and so on) are similarly migrated. Doing so may lead to unpredictable results on CONTROL-Ms which have not been migrated.

### **CONTROL-M** interface to the IBM Health Checker

When the CONTROL-M interface to the IBM Health Checker facility is enabled, the CONTROL-M monitor communicates with IBM Health Checker to run appropriate checks to ascertain the status of various CONTROL-M components, subtasks and repositories. The CONTROL-M monitor then alerts the user to any potential CONTROL-M problems via Health Checker reporting facilities. For more information on the IBM Health Checker, see the *IBM Health Checker for z/OS User's Guide* (for z/OS 1.8 or above).

The following checks are available:

- Job processing delay reports when one of various CONTROL-M components or subtasks (Submitter, Selector, Spyer) are not responding or experiencing undue delays during job processing.
- Active Jobs File (AJF) status exception messages are issued when the CONTROL-M AJF utilization threshold has been reached or exceeded.
- CTMPARM synchronization reports inconsistencies between the values set in the CTMPARM member of the IOA PARM library and the in-memory values of the CTMPARM parameters.
- DORMANT jobs check exception messages are issued, when one or more jobs are sitting in the Active Jobs file (AJF) for more than the defined number of days.

The CONTROL-M interface to the IBM Health Checker is controlled by parameters set in the CTMPARM member of the IOA PARM library. These parameters specify whether to enable the health checker interface, whether to enable each of the health checks, and the time interval at which the health checks should be run. All of these parameters may be dynamically changed without restarting the CONTROL-M monitor. For more information on setting and modifying these parameters, see the *INCONTROL for z/OS Installation Guide*.

# **CONTROL-M Support for MAINVIEW Batch Optimizer**

CONTROL-M provides scheduling support for jobs that use pipes at sites that have MAINVIEW Batch Optimizer (MVBO)/Job Optimizer Pipes installed. A pipe is a processor storage buffer that enables data to be passed between applications without using DASD or tape.

MVBO/Job Optimizer Pipes uses pipes to replace sequential job processing with parallel processing wherever feasible. Jobs and job steps normally run sequentially because they depend on data files that become available only after the application that creates them completes execution. When pipes are used, an application does not

need to finish running before the data it generates is available to other applications. This significantly reduces I/O operations and delays, and speeds up processing, because pipes enable movement of data using processor storage instead of writing and reading data to and from external storage.

CONTROL-M scheduling support for MVBO/Job Optimizer Pipes consist of the following components:

- job scheduling definition support
- enhanced runtime scheduling algorithm

These are described in the following paragraphs.

# **Job Scheduling Definition Support**

A PIPE statement can be specified in the CONTROL-M job scheduling definition for each pipe accessed by the job. Each PIPE statement contains the pipe (data set) name. The job scheduling definition of a participant includes a PIPE statement for each pipe accessed by the job.

### **Enhanced Runtime Scheduling Algorithm**

Jobs sharing a pipe are called "pipe participants." CONTROL-M recognizes each set of interrelated pipes and participants as a single, comprehensive unit called a Collection. All pipe participants are submitted concurrently, after verification that all required resources, such as prerequisite conditions or Quantitative resources, are available. This method ensures that participants do not wait for other participants to start executing, for example, at synchronization points.

For more information, see "MAINVIEW Batch Optimizer Considerations" on page 821

### Online User Interface to CONTROL-M

Until now, we have seen how CONTROL-M automates the production environment and we have discussed a number of available facilities that enhance the functionality of CONTROL-M.

However, as mentioned earlier, CONTROL-M provides an online user interface that enables the user to:

- interface with most of the previously described facilities
- intervene in the process of production management
- immediately access up-to-date information from the production environment

The online user interface is provided through online facilities that are accessed through the IOA Primary Option menu.

Certain online facilities are unique to CONTROL-M, and other facilities are shared by many or all products.

All IOA and CONTROL-M online facilities are discussed in detail in Chapter 2, "Online Facilities." They are all outlined briefly on the following pages.

#### **NOTE**



Your INCONTROL administrator can limit the options displayed on a user-by-user basis and can modify option numbers and customize option descriptions. Default options are discussed in this overview.

# **Scheduling Definition Facility**

The CONTROL-M Scheduling Definition facility is accessed through option 2 of the Primary Option menu. It is the main online facility for creating, defining, modifying, and deleting

- scheduling tables
- job scheduling definitions

In addition, this facility can be used to

- edit the JCL of a job
- produce a job (scheduling) plan
- display job statistics
- copy a job definition
- graphically display a job flow of the jobs in a table
- manually order or force jobs

#### NOTE -



Ordering places the requested job in the Active Jobs file only if its basic scheduling criteria are met. Forcing places the requested job in the Active Jobs file regardless of its basic scheduling criteria.

# Active Environment (Status) Screen: Online Tracking and Control Facility

The Online Tracking and Control facility is accessed through option 3 of the IOA Primary Option menu. It is the main user interface to the monitoring of the jobs scheduled for the day. This facility consists of a number of screens, each providing the user with relevant information and options.

The main screen of this facility is the Active Environment screen. (Prior to version 6.0.00, this screen, which displays the status of each job order in the Active Jobs file, was referred to as the Status screen.) All screens and windows available in the Online Tracking and Control facility are accessed through the Active Environment screen. In the Online Tracking and Control facility, you can perform the following functions:

- view the status of each job order in the Active Jobs file
- place a job in HELD status or free a HELD job
- delete a job order
- obtain a statistical overview of the status of jobs in the Active Environment screen
- see why a job in the Active Jobs file has not been submitted. If job submission is held up due to missing prerequisite conditions, you can optionally add those conditions manually
- display the Log file of a job to view all messages issued for the job
- zoom in on the parameters of a job order

This includes not only the job scheduling definition parameters, but also parameters determined by the CONTROL-M monitor at runtime. Manual update of some of these parameters for the job order is permitted.

- view the documentation of a job
- add notes to a job, for example, to document actions that were taken
- confirm the scheduling, rerun, or restart (if CONTROL-M/Restart is active), of a
  job that has been defined as requiring manual confirmation
- §Restart§ view the execution history of all orders of a job, and view the job order sysouts
- view the accumulated statistics of job executions
- view the list of job dependencies for a specific job, that is, the predecessor and successor jobs of the selected job, and perform manual job flow adjustment, such as priority adjustment

You can filter which jobs in the Active Jobs file are displayed in the Active Environment screen.

# **CMEM Rule Definition Facility**

The CMEM Rule Definition facility is accessed through option C of the INCONTROL Primary Option menu. CMEM rules enable CONTROL-M to respond to external events. The CMEM Rule Definition facility is an online facility that enables the user to create, define, modify and delete

- CMEM rule tables
- CMEM rules

The user can load rule tables to memory from the CMEM Rule Definition facility. Rule tables can also be loaded to memory by an operator command.

### **IOA Conditions/Resources Screen**

The IOA Conditions/Resources screen is accessed through option 4 of the IOA Primary Option menu. It displays information from the IOA Conditions file, which contains the list of all existing prerequisite conditions, and the CONTROL-M Resources file, which contains the list of Quantitative resources and Control resources. The IOA Conditions/Resources screen enables the user to

- view IOA prerequisite conditions
- view CONTROL-M Quantitative resources
- add or delete prerequisite conditions and/or resources
- change the available quantity of Quantitative resources

### **IOA Log Screen**

The IOA Log screen, accessed through option 5 of the IOA Primary Option menu, displays the IOA Log file. The IOA Log file contains messages that record every significant event in the life of all jobs or started tasks, rules, missions, and other functions that are under the control of IOA products. This includes messages generated for normal processing, such as job submitted, error conditions (if any) encountered during processing, and messages directed to the Log file from the SHOUT facility.

The user can filter IOA Log file contents displayed in the IOA Log screen.

### **IOA Manual Conditions Screen**

The IOA Manual Conditions screen is accessed through option 7 of the IOA Primary Option menu. It displays the IOA Manual Conditions file, which contains the list of prerequisite conditions that must be added manually. These are IN conditions that are required by scheduled jobs but are not added by scheduled jobs, that is, these conditions are not listed as OUT or DO COND conditions in the Active Jobs file.

These conditions fall into the following categories:

- conditions that are never automatically added by scheduled jobs because manual confirmation is always desired, for example, TAPE-ARRIVED
- conditions that are normally added automatically by scheduled jobs, but the jobs that add them are not scheduled

For the conditions listed in the Manual Conditions screen to be added to the IOA Conditions file, manual intervention is required.

The Manual Conditions list is described in Chapter 6, "Selected Implementation Issues."

The IOA Manual Conditions screen enables the user to:

- view the list of Manual Conditions
- select and add listed conditions, as desired, to the IOA Conditions file

### **IOA Calendar Facility**

The IOA Calendar facility is accessed through option 8 of the IOA Primary Option menu. IOA calendars allow definition of common scheduling patterns that simplify the entering of basic scheduling criteria in job scheduling definitions.

The IOA Calendar facility enables the user to create, define, modify and delete IOA calendars.

### **Online Utility Screens (Under ISPF)**

When CONTROL-M and other INCONTROL products (if any) are active under ISPF, a number of utilities and facilities can be activated online. The IOA Online Utilities menu is accessed through option 6 of the IOA Primary Option menu (under ISPF). The IOA Online Utilities menu displays available utilities from which the desired utility or facility can be selected.

# CONTROL-M Concepts

Having discussed CONTROL-M from a functional viewpoint, and having briefly outlined the online user interface to CONTROL-M, it is now worthwhile to discuss certain important concepts in CONTROL-M functioning.

# **IOA Core and CONTROL-M Repository**

A differentiation is made between files belonging to a particular INCONTROL product such as CONTROL-M, and IOA files that are shared among INCONTROL products.

Shared IOA files are collectively referred to as the IOA Core. The IOA Core consists of the following files:

**Table 8 IOA Core Files** 

File	Description
IOA Log file	File in which all events related to job processing are recorded.
IOA Conditions file ^a	File that lists the available conditions identified and tracked by the CONTROL-M monitor.
IOA Manual Conditions file	File listing prerequisite conditions that must be added manually, that is, prerequisite conditions required by jobs that have been ordered to the Active Jobs file and which are not automatically added by other jobs in the Active Jobs file.
IOA Calendar tables	Files containing IOA calendar definitions.
Dynamic Destination table	File containing a list of destinations for messages issued by the IOA Shout facility.
Mail Destination table	File containing a list of mail destinations for messages issued by the IOA Shout facility.
	rticular INCONTROL product are called the repository of that DL-M Repository consists of the following files:
Active Jobs file	File used to hold copies of the job scheduling definitions of those jobs that have been ordered that working day.
CONTROL-M Resources file ^a	File that lists the available resources identified and tracked by the CONTROL-M monitor.
Scheduling tables	Files containing job scheduling definitions.
CMEM Rule tables	Files containing CMEM rule definitions.
Job Statistics file	File containing the execution statistics of all jobs.
Job Network file	File containing dependency information about the jobs in the Active Jobs file.
History Jobs file	File containing jobs that ended OK or expired.
Journal file	File containing data about changes to the CONTROL-M Active Jobs file, the CONTROL-M Resources file, and the IOA Conditions file ^a , and which can be used for Restoration purposes.

^a Prior to version 6.0.00, conditions and resources were stored in a single file, the IOA Conditions/Resources file.

# **Date Definition Concepts**

INCONTROL recognizes the following types of date definitions. Depending on the INCONTROL product, either all of them, or some of them, are relevant. All these types are relevant for CONTROL-M:

**Table 9** Date Definition Types

<b>Date Definition</b>	Description
System date	Date as supplied by the operating system. This date must be the actual calendar date starting and ending at midnight.
Working date	Many sites do not use midnight as the formal time for changing to a new date. A site, for example, may determine that all processing performed between the hours of midnight and 6:00 a.m. "belongs to" the previous day. In this case, the installation working date at the site changes at 6:00 a.m., not at midnight.  The working date, that is, the time at which the date changes at the site, is defined in the CONTROL-M installation parameters. New Day processing generally begins at the start of the new working date.
Original scheduling date	Job orders and prerequisite conditions managed by CONTROL-M are assigned an original scheduling date, which is referred to as ODATE. For the full implications of using ODATE, see "ODATE" on page 153. For details of the enhanced meaning of ODATE as of version 6.1.00, see "Enhanced Definition of ODATE" on page 64.

### **Example 1**

A computer is down for repairs on February 2nd and 3rd. When it is brought up on February 4th, a two-day backlog of jobs must be run in addition to the jobs of the current day.

When the New Day procedure scans scheduling tables on February 4th, it places job orders in the Active Jobs file for all three days. Jobs that ought to have run on February 2nd are assigned an ODATE of February 2nd, jobs for February 3rd are assigned an ODATE of February 3rd, and so on.

In this manner, each job is executed as if it had run on the working date on which it was originally scheduled.

### Example 2

ODATES are calculated according to the working date, and not the calendar date.

If you define a job to run on 5 December at 3 A.M., and the working day begins (and the New Day procedure operates) at 5 A.M., the job will not run until 3 A.M. on 6 December, because that is still part of the working day of 5 December.

### **Enhanced Definition of ODATE**

As of version 6.1.00, ODATE has an enhanced definition. ODATE can also be one of the runtime criteria, such as IN conditions, that must be satisfied before a job can be submitted. Runtime criteria are explained in "Automated Job Submission" on page 47 and "Monitoring of Resources" on page 48.

While prior to version 6.1.00 the ODATE was only a VALUE date, it can now be both a RUN date and a VALUE date.

### **ODATE** with the Attribute VALUE

In most cases, ODATE by default has the attribute VALUE. This means that it is a VALUE date, and is not one of the runtime criteria.

When ODATE has the attribute VALUE, it has the following characteristics:

- ODATE is a logical date that is used by CONTROL-M when adding jobs to the Active Jobs file for execution. The ODATE is assigned to a job by manual order or by operation of the New Day procedure.
- The ODATE is a 24 hour period. It begins at the New Day time. During the 24 hour period that follows that New Day time, all job scheduling is based on the ODATE, which corresponds to the calendar date at that New Day time, rather than the calendar date at the time when the job runs.
- The ODATE can coincide with, precede, or follow the calendar date. If no value is set for the DAYTIMEM parameter in the CTMPARM member, the ODATE coincides with the calendar date. If the DAYTIMEM parameter is set using a (Minus) sign, the ODATE precedes the calendar date by the number of hours and minutes specified in that parameter. If a + (Plus) sign is used, the ODATE follows the calendar date in a similar manner.

For more information on the DAYTIMEM parameter, see the description of operational parameters in the CONTROL-M chapter of the *INCONTROL for z/OS Installation Guide*.

- When a job is eligible to be ordered on an ODATE, it is placed in the Active Jobs file, and is immediately eligible for submission as soon as all its runtime criteria, such as TIME FROM and TIME UNTIL, have been met.
- When the end of the ODATE arrives, the New Day procedure may remove jobs with that ODATE from the Active Jobs file, depending on the setting of the MAXWAIT parameter of the specific job. Jobs removed in this way cease to be eligible for submission.

#### **ODATE** with the Attribute RUN

Although by default ODATE has the attribute VALUE, it may also have the attribute RUN, if either set by the user, or the New Day procedure. In such cases, a job can only run when its ODATE is the same as, or after, the CONTROL-M logical date. In other words, the ODATE becomes a runtime criteria.

In this context, runtime criteria are the criteria that determine the eligibility "window" for the submission of the job, that is, the period of time during which the job can be submitted. This eligibility window is determined by the ODATE and the TIME ZONE parameter setting.

For information on changing the attribute of ODATE from VALUE to RUN, see the description of the Time Zone feature in the *INCONTROL* for z/OS Administrator Guide, and the description of the CTMJOB utility in the *INCONTROL* for z/OS Utilities Guide.

### **Date Standards and Date Field Formats**

Date standards and date field formats use either Gregorian or Julian dates.

# **Gregorian Dates**

Gregorian dates are indicated in the guide by the following symbols:

Table 10 Gregorian Date Notation

Symbol	Description
dd	Day of the month (01 – 31)
mm	Month (01 – 12)
уу	Last two digits of the year ^a
уууу	Four digits of the year

^a If the last two digits in the specified year are a number less than 56, IOA presumes that the year is in the 21st century; for example, if *yy*=15, the year 2015 would be presumed. Otherwise, IOA presumes that the year is in the 20th century; for example, if *yy*=80, the year 1980 would be presumed.

Whether a field holds a 4-character date (month and day), a 6-character date (month, day and 2-digit year) or an 8-character date (month, day and 4-digit year) depends on the field definition. However, the format of the 4-character, 6-character or 8-character date depends on the date standard defined during installation.

INCONTROL products support three date standards for Gregorian dates. Each standard has an 8-character format, a 6-character format and a 4-character format. Only one Gregorian date standard is defined at any site.

These supported Gregorian date standards are described in the chart below.

**Table 11 Supported Gregorian Dates** 

Standard	4-Character Date	6-Character Date	8-Character Date
MDY	mmdd	mmddyy	mmddyyyy
DMY	ddmm	ddmmyy	ddmmyyyy
YMD	mmdd	yymmdd	yyyymmdd

### **Julian Dates**

Julian dates (also supported by INCONTROL products) are indicated in the guide by the following symbols:

Table 12 Julian Date Notation

Symbol	Description
jjj or ddd	Day of the year (001 – 365 or 366, as appropriate for the year)
уу	Last two digits of the year
уууу	Four digits of the year

Julian date fields have either three, five, or seven characters. Whether a Julian date field holds a 3-character date (day of year only), 5-character date (day of year and 2-digit year) or a 7-character date (day of year and 4-digit year) depends on the field definition. However, the format of the date depends on the installation-defined date standard.

For example, the Julian date for the calendar date of 28 February 2001 would be represented in jij or ddd format as 059, in yyjjj or yyddd format as 01059, and in yyyyjjj or yyyyddd format as 2001059.

# **Job Ordering and Job Forcing**

Job ordering is the placing of a job scheduling definition in the Active Jobs file when the basic scheduling criteria of the job are satisfied.

Most production jobs are automatically ordered during New Day processing. However, jobs can be manually ordered, as well.

Job forcing is the placing of a job scheduling definition in the Active Jobs file regardless of the basic scheduling criteria of the job.

Although any job can be forced, job forcing is generally requested for special purpose, or exception, jobs that are not normally scheduled:

- Jobs can be automatically forced as part of the post-processing of another job. For example, a particular job may be required only if a certain other job abends. In this case, it is forced during the post-processing for the abended job.
- Jobs can also be forced manually. For example, a routine job that is generally
  ordered automatically according to its scheduling criteria can be manually forced,
  if required, on a day it is not normally scheduled.

### **Rerun and Restart**

Rerun and restart are two distinct, though related, concepts.

Rerun is the re-execution of a job from the beginning. Job rerun is a CONTROL-M feature.

Restart is the re-execution of a job from a predefined step. Restart is usually performed from the step that failed, although it can be performed from an earlier step, if necessary. Restart utilizes the successful steps from the failed job execution, thereby limiting the amount of processing required to complete successful job execution. This results in lower CPU overhead, and can make a big difference in the timely completion of processing.

A basic MVS restart capability is available, and is described in "OUT: Post–Processing Parameter" on page 562. BMC Software do not recommend this method. This type of restart starts execution of the job from the failed step. However, no auxiliary restart functions are performed.

**§Restart§** By contrast, at sites in which CONTROL-M/Restart is installed, restart under CONTROL-M/Restart is available. In addition to performing restart from the desired step, with the capability of automatic step rollback when necessary, CONTROL-M/Restart automatically performs auxiliary restart functions. These include the cataloging and scratching of data sets, prevention of NOT CATLGD 2 errors, and so on.

**§Restart§** Instructions for rerun and restart can be defined in the job scheduling definition. Rerun is defined with the DO RERUN statement. Restart is defined with the DO IFRERUN statement. They can be defined to be performed automatically or to be performed upon manual confirmation. For more information, see "DO RERUN: Post–Processing Parameter" on page 462, "§Restart§DO IFRERUN: Post–Processing Parameter" on page 447, and the *CONTROL-M/Restart User Guide*. **§Restart§** 

### **Order ID**

CONTROL-M can handle multiple orders of the same job. To distinguish between the job orders, CONTROL-M assigns each job order a unique order ID. Therefore, it is not uncommon to see the same job name with multiple order IDs, each representing a different job order, in the Active Environment screen.

### **SYSDATA**

SYSDATA is the term used to designate the data in three job sysout data sets:

- job log (console messages)
- expanded JCL
- system output messages

SYSDATA data sets are usually produced for each execution of a job or started task. However, not all three data sets are necessarily present in all cases. For example, in JES2, if a job is canceled by the operator before execution, the system output messages data set might not be produced.

For jobs, the output class for this data is defined by one of the following:

- MSGCLASS parameter on the job card, which is added or overwritten by CONTROL-M during job submission
- JCL job-level //OUTPUT statement using the JESDS subparameter
- default values defined in JES initialization parameters
- for started tasks, in JES initialization parameters

When CONTROL-M/Restart is installed, it uses the SYSDATA to analyze the execution of a job order, beginning with the archived SYSDATA of the most recent non-restarted run.

# **Handling of Job Groups**

Normally, the handling of each job in a table is independent of the handling of the other jobs in the table. Each job is handled according to the criteria specified in its own job scheduling definition.

However, the Scheduling Definition facility also supports the handling of jobs as a group. Such jobs are defined in a special scheduling table, called a Group scheduling table. Each Group scheduling table has a special job scheduling definition, called a Group Entity. Group handling criteria for the entire group of jobs are specified in this Group Entity. These include:

Table 13 Group Handling Criteria

Criteria	Description
Basic Scheduling criteria	Scheduling criteria to be applied to jobs in the group.
Runtime Scheduling criteria	Required runtime criteria for all scheduled jobs in the group.
Post Processing actions	Actions to be performed when all scheduled jobs in the group have finished executing with the appropriate status.

### **Dynamic Group Insert**

When a group is ordered, the group entity and some or all of its jobs are placed on the Active Jobs File. The Dynamic Group Insert facility makes it possible to insert additional jobs belonging to this group into the group entity that is already on the Active Jobs File.

The additional jobs must be jobs that belong to the group. They may be either or both of the following:

- jobs that were not scheduled at the current time
- additional copies of jobs that are already in the Active Jobs File

For more information about using the Dynamic Group Insert facility, see the description of the job ordering facility CTMJOB in the CONTROL-M Utilities chapter of the *INCONTROL for z/OS Utilities Guide*.

# **Prerequisite Conditions**

The prerequisite condition concept is one of the key concepts of CONTROL-M production control.

Prerequisite conditions enable the establishment of job dependencies and, when a job normally requires manual intervention, such as determination that a cartridge arrived on-site, ensures that the manual conditions are satisfied before the job is submitted.

A prerequisite condition is a user-defined, descriptive name given to a certain situation or condition. Prerequisite conditions can be specified in any of three types of statements in a job scheduling definition:

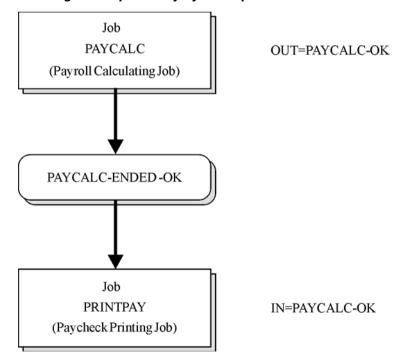
**Table 14** Prerequisite Condition Statements

Statement	Description
IN statements	These statements must be satisfied (that is, the prerequisite condition must exist) before the job can be submitted.
OUT statements	These statements are performed, that is, the prerequisite conditions are added or deleted, only when the job ends OK.
DO COND statements	Whether these statements are performed (that is, the prerequisite conditions are added or deleted) depends on the execution results of the job.
	DO statements in a job scheduling definition accompany ON statements. The ON statements define step and code criteria. If the specified code criteria are satisfied for the specified steps, the accompanying DO statements are performed.

In its most basic form, a prerequisite condition is defined in an IN statement in one job, and as an OUT (or DO COND) statement in another job. This makes the execution of the one job dependent on the execution of the other job.

### **Example**

Figure 1 Establishing Job Dependency by Prerequisite Conditions



Payroll-calculating job PAYCALC must be run before Payroll-check-printing job PRINTPAY. To create the necessary job dependency, a prerequisite condition is defined as follows:

- Prerequisite condition PAYCALC-ENDED-OK is defined as a runtime scheduling criteria in the job scheduling definition for job PRINTPAY.
- Prerequisite condition PAYCALC-ENDED-OK is defined as a post-processing parameter for job PAYCALC, only when job PAYCALC terminates successfully.

Because the condition required by job PRINTPAY is not created unless job PAYCALC terminates successfully, the dependency of job PRINTPAY on job PAYCALC is established.

Job dependencies do not have to be as simple as the above example illustrates. An almost unlimited number of conditions and job dependencies can be created:

- jobs can be dependent on more than one prerequisite condition
- jobs can add and/or delete more than one prerequisite condition
- the same prerequisite condition can be added by more than one job (caution must be used)
- the same prerequisite condition can be used as an IN condition for more than one job

In Group scheduling tables (described in "Handling of Job Groups" on page 68), prerequisite conditions can be defined as IN, OUT and/or DO COND conditions in the Group Entity. In this case, they apply to the entire set of scheduled jobs.

## **Prerequisite Condition Dates**

IN, OUT, and DO COND statements provide a field for specifying a date to accompany each prerequisite condition. An OUT or DO COND prerequisite condition that is added with a particular date cannot satisfy the same IN prerequisite condition if the IN statement specifies a different date.

### **Example**

JOB_A and JOB_B each run daily, and JOB_B is dependent on JOB_A.

(JOB_A has prerequisite condition JOB_A_ENDED_OK as an OUT condition, and JOB_B has the same condition as an IN condition.)

The date associated with a condition is important because it is absolutely necessary that, on a given day, JOB_B not be triggered by an occurrence of the condition JOB_A_ENDED_OK from a previous day.

Certain Date keywords can be specified in place of, and resolve to, actual date values. For example, keyword ODAT is automatically replaced by the original scheduling date of the job.

Another important keyword for use in place of an actual date is STAT. STAT is used as a date reference for conditions that are static, that is, not date-dependent.

For example, condition IMS_ACTIVE is added when IMS is brought up, and only deleted if IMS is brought down. The date of the condition is irrelevant to jobs requiring that condition. Therefore, this condition would be referenced with a date value of STAT.

#### NOTE —



Before STAT was introduced, date 0101 was recommended to be used in conditions that were not date-dependent. Unlike 0101, STAT is not a date, and it operates differently. Always use STAT when defining conditions that are not date-dependent.

# **Deleting Conditions**

The last job to require a particular prerequisite condition, that is, in an IN statement, can also mark that condition for deletion, that is, in an OUT statement. The deletion of unnecessary conditions can serve the following purposes:

It can eliminate unnecessary clutter from the IOA Conditions file and the CONTROL-M Resources file, and the IOA Conditions/Resources screen.

When dependent jobs are scheduled multiple times each day, it can prevent the execution of the earlier scheduled "predecessor" job from incorrectly causing the submission of the later scheduled "successor" job.

### **Conditions Requiring Manual Intervention**

Prerequisite conditions can be used to ensure that a required manual operation has been performed. The following example illustrates such a condition.

### **Example**

The job scheduling definition of JOB-A specifies prerequisite condition TAPE-ARRIVED as runtime scheduling criteria. When the operator sees that JOB-A is waiting for this condition to be satisfied, the operator can verify that the required external tape has arrived at the site, and then use the online facility to manually add the condition to the IOA Conditions file (through the Manual Conditions screen, the IOA Condition/Resources screen, or the Why screen). The job can then be submitted by CONTROL-M.

### **Maybe Jobs**

In some cases, job dependencies created by prerequisite conditions are desired only if the predecessor jobs are scheduled. If the predecessor jobs are not scheduled, ignore the dependencies.

Such dependencies are called Maybe dependencies, and the unscheduled predecessor jobs that are ignored if they are not scheduled are called Maybe jobs. Conditions set by unscheduled Maybe jobs appear in the Manual Conditions file.

The Manual Conditions file and the handling of Maybe jobs are discussed in Chapter 6, "Selected Implementation Issues."

# **Quantitative and Control Resources**

To prevent bottlenecks and help guarantee successful execution of jobs, CONTROL-M provides tools to ensure that a job is not submitted for execution until all resources required by the job are available.

### **Quantitative Resources**

Specification of Quantitative resource requirements for a job provides a solution for the allocation of quantitative computer resources, such as cartridge drives, CPU utilization, and database access-rate. It increases computer throughput by controlling access to these resources, thus preventing execution bottlenecks.

CONTROL-M maintains a continuously updated status of the Quantitative resources of the site in the CONTROL-M Resources file.

When a Quantitative resource is specified for a job, CONTROL-M determines if a sufficient quantity of the specified resource is available before submitting the job. When the job is submitted, the specified quantity of resource is allocated to that job and is unavailable to other jobs. When the job finishes executing, the resource is made available to other jobs.

The quantity of each resource that is available in the data center is specified using CONTROL-M utilities. An authorized user can dynamically change these quantities manually from the IOA Conditions/Resources screen.

### **Control Resources**

Specification of resource control requirements for a job provides a solution for the problem of resource sharing between different jobs. The mode (Exclusive or Shared) in which a resource is required by a job can be specified.

For example, a job that reads a database without performing updates can access the database in Shared mode; any other job requiring read-only access to the database can access the database at the same time. Conversely, a job that updates the database may require Exclusive control of the database at the time of update such that no other jobs can share the database.

In the example just presented, the database can be defined as a Control resource, and the type of control required by the job (Exclusive or Shared) can be specified for the resource.

CONTROL-M considers the mode of resource usage required when allocating Control resources and prevents jobs whose resource usage is incompatible from executing simultaneously.

# **Job Priority**

The job scheduling definition may include a specification of an internal priority for the job. When competing for the same resource, jobs with higher priority take precedence over jobs with lower priority. Users can also assign a "critical path" priority to jobs that must be submitted with the least delay possible. A job with critical path priority is allocated required resources as the resources become available. When all its required resources are available, the job is submitted.

Noncritical jobs are not allocated resources until all required resources are available at the same time.

# **Automatic Job Flow Adjustment**

Predecessor and successor job flows are established through the use of prerequisite conditions that are defined in the job scheduling definition. Successor and predecessor jobs are identified as either "immediate" or "eventual," relative to a specified job:

- An immediate predecessor and successor relationship exists between jobs when one job is directly dependent on prerequisite conditions added by the other job.
- An eventual predecessor and successor relationship exists between jobs if their dependency is indirectly established through a "chain" of immediate predecessor and successor jobs.

From the network of predecessor and successor jobs, critical paths can be identified. A critical path is a chain of jobs that must be executed in their appropriate sequence in order for a specified job to run. A job can have more than one critical path, if different jobs set the same OUT condition, or if a job has OR logic in its IN conditions.

The Job Dependency Network screen, accessed through the Active Environment screen, enables you to view the network of predecessor and successor jobs for a specified job and determine the critical paths for the job.

Although it is prerequisite conditions that define predecessor and successor job relationships, the actual job flow along a critical path can be greatly impacted by the following runtime scheduling criteria in the job scheduling definition:

**Table 15** Runtime Scheduling Criteria

Criteria	Description	
PRIORITY	As mentioned earlier in "Job Priority," a PRIORITY value affects the selection order of the job (relative to other jobs).	
DUE OUT	The date and time by which the job must finish executing.	

In some cases, it may become desirable to adjust the priorities or due out dates and times of certain job orders.

### **Examples**

- A high priority successor job is waiting for the submission (and completion) of a lower priority predecessor job.
- A predecessor job cannot terminate early enough for a successor job to terminate by the due out date and time of the successor.

Both types of job flow adjustments can be requested from the Job Dependency Network screen:

#### ■ Priority Propagation

The priority value of each non-Held predecessor and successor job is checked and (if necessary) modified so all jobs in the chain have a priority, and no job has a lower priority than any of its successor jobs.

### ■ Deadline Adjustment

Starting with the latest eventual successor job in the job flow, the anticipated elapsed time (that is, anticipated execution time) is subtracted from the DUE OUT date and time to determine DUE OUT date and time of the immediate predecessors of that job.

This process of subtracting elapse times of a job to determine the DUE OUT date and time of the immediate predecessor jobs are repeated until the DUE OUT date and time of the initial or current job is determined.

- If the user entered an ELAPSE time value in the Online Tracking and Control facility Zoom screen, this value is used for the above calculation.
- If the user did not enter an ELAPSE time value, the anticipated elapse time is determined by the average runtime taken from the CONTROL-M Statistics file.

### Note the following points:

- By subtracting the ELAPSE time of a job from its DUE OUT date and time, the CONTROL-M monitor calculates a DUE IN date and time (that is, the date and time by which the job must be submitted) for each job. The DUE IN time, ELAPSE time, and DUE OUT time are also displayed in the Job Dependency Network screen.
- The ELAPSE time, DUE OUT date and time, DUE IN date and time, and PRIORITY values for a job are also displayed in the Zoom screen, which is accessed through the Active Environment screen.
- DUE OUT date and time, ELAPSE time and PRIORITY values can also be manually modified in the Zoom screen, but it is recommended that this not be done, and that automatic job flow adjustment be requested instead.

Deadline adjustment will work correctly only if all the jobs have the same time zone, or all the jobs have no time zone.





# **Online Facilities**

# This chapter includes the following topics:

Overview	80
IOA Features	80
General IOA Features	80
IOA Entry Panel	84
IOA Primary Option Menu	85
Multi-Screen Control	90
Fast Exit from the IOA Online Facility	91
Screen Layout	92
Commands and PFKeys	93
Online Help	99
AutoRefresh Mode	. 100
IOA Under ISPF	. 101
IOA Editor	. 102
IOA SET Command Panel	. 105
IOA TSO Command Processor Screen	. 108
Scheduling Definition Facility	. 110
Entry Panel	
Table List screen	
Job List Screen	. 123
Job Scheduling Definition Screen - Defining Schedules	. 127
Exiting the Scheduling Definition Facility	
Ordering (Scheduling) Jobs	. 150
Copying Jobs to Another Table	
Deleting Tables	
Condition Inheritance	. 159
Displaying Graphic Jobflow	. 162
Displaying a Job Scheduling Plan	. 164
Tracking and Control Facility	
Active Environment Screen	. 167
Global View Screen	. 199
View Graph Screen	. 201
Why screen	. 205
Deleting a Job	
Log Screen	. 212
=	

Zoom Screen	213
Confirm Scheduling Window	222
Confirm Rerun Window	222
§Restart§ Confirm Restart Window (Under CONTROL-M/Restart)	
§Restart§Rerun and/or Restart Window (Under CONTROL-M/Restart)	
Rerun Flow Job List Window	
Step List Window	
§Restart§Job Order Execution History Screen	
SRestarts Sysout Viewing Screen	
Statistics Screen	
Job Dependency Network Screen	
History Environment Screen	
Force OK Confirmation Window	
CMEM Rule Definition Facility	
Entry Panel	
Table List Screen.	
Rule List Screen	
Rule Definition Screen – Defining Rules	
Entering Comments	
Editing CMEM Rule Definitions in the Edit Environment	202 263
Exiting the CMEM Rule Definition Facility	
Deleting Tables	
Ordering CMEM Rule Tables.	
Copying Rules to Another Table	
IOA Variables Database Facility	
Entry Panel	
Values of Database Screen	
Variable Zoom Screen	
Condition and Resource Handling Facility	
IOA Manual Conditions/Resources Screen	
IOA Landard Conditions Screen	
IOA Log Facility	
IOA Colon Brillian	
IOA Calendar Facility	
Accessing the IOA Calendar Facility	
Entry Panel	
Calendar List Screen	
Year List Screen	
Copying Years to Another Calendar	
Calendar Definition Screen.	
Exiting the IOA Calendar Facility	
Utilities Under ISPF	
IOA Online Utilities Menu	
I1: Add, Check, or Delete a Prerequisite Condition	
M1: Issue a Job Order	
M2: Perform an AutoEdit Simulation	
M3: Prepare Simulation/Tape Pull List Job	
M4: Parameter Prompting Facilities	337

M5: Quick Schedule Definition	<b>360</b>
M6: End-User Job Order Interface	370
U1: Invoke DOCU/TEXT	372

# **Overview**

The Online facility is the basic means of communication between the user and CONTROL-M for z/OS.

Online job scheduling definition gives users the ability to define and modify job production parameters in the CONTROL-M production environment.

Online tracking displays the current status of all variables relating to a specific job, a group of jobs or all jobs scheduled under CONTROL-M.

Online control enables authorized users to modify variables relating to a specific job, a group of jobs or all jobs scheduled under CONTROL-M.

The following pages describe the main features available under the Online facility.

# **IOA Features**

This section discusses the IOA features common to all INCONTROL products.

### **General IOA Features**

General IOA features include:

- Customization
- **■** Environment Support
- Terminal Support
- Special Character Usage on Terminals
- **■** Color Support
- Prefixing
- Character Masking

### **Customization**

IOA screens, constants, messages, colors, commands, and PFKey definitions can be site-modified to adapt them to local needs. For further details, see the *INCONTROL* for z/OS Installation Guide.

INCONTROL products can be customized globally, that is, for the whole site, using the INCONTROL Installation and Customization Engine (ICE), according to profile variables defined during installation.

In addition, INCONTROL products can be customized to respond differently to individual users if these profile variables are specified in user profile members.

For example, depending on the setting of a variable in a particular user profile member, upon exit from a screen in which changes have been requested, this INCONTROL product may either perform the requested changes automatically or display a confirmation window before performing the changes.

Customization issues are discussed in the INCONTROL for z/OS Installation Guide.

#### NOTE -



Due to customization, the screens and examples illustrated in this guide may differ from the ones used at your site. The \$\$ACTDOC member of the IOA MSG library contains information that is useful for customizing the CONTROL-M Active Environment screen and creating and modifying display types for screens 3, 3.N, 3.G and the History Environment screen.

### **Environment Support**

The Online facility can be activated under the following environments:

- TSO (native)
- TSO/ISPF
- ROSCOE/ETSO
- CICS
- VTAM
- IMS/DC
- IDMS/DC
- COM-PLETE

Cross memory interfaces (to the Online monitor) are optional under native TSO, TSO/ISPF, and ROSCOE/ETSO. They are always used under the other environments.

There are slight differences in the operation of the Online facility under the different environments. Special notes are provided in this guide where applicable.

# **Terminal Support**

IOA supports the following models of IBM 3270 terminals:

- Model 2 24 lines, 80 columns
- Model 4 43 lines, 80 columns
- Model 3 32 lines, 80 columns
- Model 5 27 lines, 132 columns

#### NOTE



When using the IOA online facility under IMS/DC and IDMS/DC, all model types display 24 lines and 80 columns.

IOA adjusts to the screen size in order to use the maximum available data area on the screen.

# **Special Character Usage on Terminals**

In certain cases, special keyboard characters, such as \$, #, and @, are assigned special meanings. The characters specified appear on standard American terminals but may not be available on other keyboards. In addition, some special characters on your keyboard may be assigned different hexadecimal values than the ones recognized by IOA. Special keyboard mapping requirements, and a complete discussion of the conventions used in this guide, are shown in "Conventions Used in This Guide" on page 34.

### **Color Support**

When INCONTROL products are activated from a screen with extended seven-color support, they make extensive use of the color attributes of the screen. The concept of management by color is emphasized in INCONTROL screens.

Like all screen attributes, the color attribute for each field is defined externally to the program and can be locally modified by the site.

### – NOTE



IOA does not automatically recognize IMS/DC and IDMS/DC terminals as supporting extended color attributes. If your IMS/DC or IDMS/DC terminal supports extended color attributes and you want IOA to recognize this, refer to the *INCONTROL for z/OS Administrator Guide* for more information.

At this time, IOA does not support extended color attributes under COM-PLETE.

Due to ISPF characteristics, color changes cannot occur in adjacent columns but must be separated by an attribute byte without color, that is, black. Therefore, some IOA screens have a different appearance under ISPF than under other online environments, such as native TSO and CICS.

# **Prefixing**

For fields that automatically support prefixing, selection strings are always treated as prefixes. Selection is made if a segment of the text beginning with the first letter, that is, any prefix, matches the selection criteria.

### **Examples**

Assume the following names exist: A3, A4, M, M01, M03, M12, M13, M22, M23, M30, M33, M103, M135, M301.

**Table 16 Prefixing Examples** 

Entry	Matching Value	
blank	All of the above values	
A	A3, A4	
M	M, M01, M03, M12, M13, M22, M23, M30, M33, M103, M135, M301	
M1	M12, M13, M103, M135	
M13	M13, M135	

If a field supports prefixing, this fact is indicated in its description.

# **Character Masking**

For fields that support masking, mask characters function as follows:

- * represents any number of characters, including no characters
- ? represents any one character

For fields that do not automatically support prefixing, a prefix value can be specified by ending the selection string with an asterisk.

### **Examples**

Assume the following names exist: A3, M, M3, M01, M03, M13, M23, M33, M103, M435, M2243.

**Table 17 Masking Examples** 

Entry	Matching values	
*	All the above values	
M?3	M03, M13, M23, M33	
M?3*	M03, M13, M23, M33, M435	
M??3	M103	
M*3	M3, M03, M13, M23, M33, M103, M2243	
M*	M, M3, M01, M03, M13, M23, M33, M103, M435, M2243	
	Since the last character in this example is *, M is treated as a prefix.	

If a field supports masking, this fact is indicated in its description.

# **IOA Entry Panel**

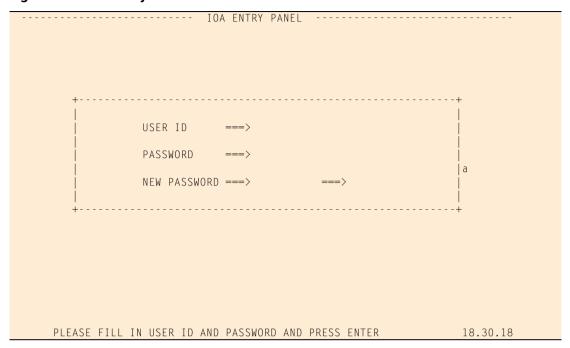
Enter the IOA Online facility according to the instructions of your INCONTROL administrator. Upon entering the IOA Online facility, the IOA entry panel may be displayed.

### NOTE -



Display of the IOA Entry Panel is optional. If your INCONTROL administrator determined that the entry panel is bypassed, the IOA Primary Option menu, which is discussed in the following section, is displayed.

Figure 2 IOA Entry Panel



Type your user ID and password and press **Enter**. If you enter a correct user ID and password, the IOA Primary Option menu is displayed.

The IOA Online facility allows three attempts to enter a valid user ID and password combination. After the third unsuccessful attempt, the program is terminated.

To change a password, type the new password twice: Once in the **NEW PASSWORD** field and once in the confirmation field to the right of the **NEW PASSWORD** field.

# **IOA Primary Option Menu**

The IOA Primary Option menu is the primary interface to functions available under the various INCONTROL products. The options displayed in the menu depend on the INCONTROL products installed at the site, and the functions and facilities that have been authorized to you.

If only CONTROL-M is installed at your site, and you are authorized to access all functions and facilities, the following screen is displayed:

#### NOTE



When the Online facility is activated as an ISPF application, option 6 is displayed as "6 UTILITIES Online Utilities." In this case, option 6 activates the Online utilities under ISPF. When the Online facility is not activated under TSO or TSO/ISPF, option 6 is inactive.

Figure 3 IOA Primary Option Menu where only CONTROL-M is Installed

OPTION ===>	IOA PRIMARY OPTION MENU	USER DATE	N22A 19.08.01
3 ACTIVE ENV. C CMEM DEFINITION 4 COND-RES 5 LOG 6 TSO 7 MANUAL COND 8 CALENDAR DEF	CTM Event Manager Rule Definition IOA Conditions/Resources Display IOA Log Display Enter TSO Command		
COMMANDS: X - EXIT, HE	LP, INFO OR CHOOSE A MENU OPTION		17.59.32

To select an option, type the option number or letters in the **OPTION** field and press **Enter**. Alternatively, for a number option, press the PFKey with the same number. For example, to select the LOG option, press **PF05/PF17**.

#### NOTE



Your INCONTROL administrator can limit the options displayed on a user-by-user basis, and can alter option numbers and customize option descriptions. Product-supplied default options are discussed in this guide.

Certain IOA commands, functions, and facilities (options) are shared by all INCONTROL products. These shared IOA commands, functions and facilities are described later in this chapter, and outlined in Table 18.

Table 18 INCONTROL Shared IOA Functions and Facilities (part 1 of 2)

Option	Function	Description
4	COND/RES	Display and update the status of the IOA Conditions file and the CONTROL-M Resources file.
5	LOG	View audit trail information about jobs, missions, and rules scheduled under the supervision of INCONTROL products.
6	TSO ^a	Perform TSO commands.
7	MANUAL COND	Display the list of prerequisite conditions that must be confirmed manually by operations personnel.
8	CALENDAR DEF	Define scheduling calendars.

Table 18 INCONTROL Shared IOA Functions and Facilities (part 2 of 2)

Option	Function	Description	
X	EXIT	Exit the Online facility.	
INFO	INFO	Display a window in the IOA Primary Option Menu. The window contains information about installed INCONTROL products. For more details on the information displayed by this command, see "IOA Version Information" on page 89.	

^a When the Online facility is activated as an ISPF application, option 6 is displayed as "6 UTILITIES Online Utilities." In this case, option 6 activates the Online utilities under ISPF. When the Online facility is not activated under TSO or TSO/ISPF, option 6 is inactive.

#### NOTE



Entering =1 in the command line of any other screen returns you to the IOA Primary Option Menu that is displayed at your site.

The following commands, functions, and facilities (options) are applicable to CONTROL-M:

Table 19 CONTROL-M Functions and Facilities

Option	Function	Description
IV	VARIABLE DATABASE	Define, display, and update IOA Database variables.
2	JOB SCHEDULE DEF	Define and modify job production parameters.
3	JOB STATUS	Display and update status of jobs scheduled under CONTROL-M.
С	CMEM DEFINITION	Define and modify CMEM rules.

The following IOA Primary Option menu is displayed at sites supporting all currently available INCONTROL mainframe products.

#### – NOTE



Option OK (KOA Recorder facility) is available only under IOATSO, and not under IOAISPF or IOAMON.

When the IOA online facility is activated as an ISPF application, option 6 is displayed as "6 UTILITIES Online Utilities." In this case, option 6 activates the Online utilities under ISPF. When the online facility is not activated under TSO or TSO/ISPF, option 6 is inactive.

Figure 4 IOA Primary Option Menu when all INCONTROL Products are Installed

OPTION ===>	IOA PRIMARY OPTION MENU	USER NO6
IOA	CONTROL-D/V	CONTROL-O
4 COND-RES 5 LOG 6 TSO 7 MANUAL COND 8 CALENDAR DEF IV VARIABLE DATABASE	A MISSION STATUS M MISSION DEF R REPORT DEF T RECIPIENT TREE U USER REPORTS F PC PACKET STATUS DO OBJECTS	OM MSG STATISTICS OS RULE STATUS OL AUTOMATION LOG OA AUTOMATION OPTS
CONTROL-M & CTM/Restart	CONTROL-M/Analyzer	CONTROL-M/Tape
2 JOB SCHEDULE DEF 3 ACTIVE ENV. C CMEM DEFINITION	BM MISSION DEF	TP POOL DEFINITION TV VAULT DEFINITION
COMMANDS: X - EXIT, HELP,	INFO OR CHOOSE A MENU OPT	TION 16.20.21

#### NOTE -



Entering =1 in the command line of any other screen returns you to the IOA Primary Option Menu that is displayed at your site.

For a description of the options for other INCONTROL products, see the user guides of the respective products.

Additional options available on the IOA Primary Option Menu when operating CONTROL-M with other INCONTROL products are listed in Table 20.

Table 20 IOA Primary Option Menu Options (part 1 of 2)

Option	Name	Description
A	MISSION STATUS	Display and update active mission status.
M	MISSION DEF	Define migration, printing, backup, and restore missions.
R	REPORT DEF	Define decollating missions (including indexing).
T	RECIPIENT TREE	Display and update the recipient tree.
U	USER REPORTS	Display and update the status of user reports. View reports online.
F	PC PACKET STATUS	Display the status of reports (packets) scheduled for transfer from the mainframe to a PC.

Table 20 IOA Primary Option Menu Options (part 2 of 2)

Option	Name	Description		
DO	OBJECTS	Manage CONTROL-D objects.		
	Note: Options A, M, R, T, U, F, and DO are available only at sites where CONTROL-D or CONTROL-V are installed.			
BB	BALANCING STATUS	Display and update the status of active balancing missions.		
BM	MISSION DEF	Define balancing missions.		
BV	DB VARIABLE DEF	Define, display and update Database variables.		
BR	RULE DEFINITION	Define balancing rules.		
BA	RULE ACTIVITY	Display rule activity and the result of invoking CONTROL-M/Analyzer rules.		
	Note: Options BB, BM, BV, BR, and BA are available only at sites where CONTROL-M/Analyzer is installed.			
OR	RULE DEFINITION	Define rules.		
OM	MSG STATISTICS	View message statistics.		
OS	RULE STATUS	View Rule Status screen.		
OL	AUTOMATION LOG	Display commands, messages and/or traces.		
OA	AUTOMATION OPTS	Display available operator productivity tools.		
OC	COSMOS STATUS	Display or modify the status of COSMOS-controlled objects and databases.		
OK	KOA RECORDER	Record VTAM scripts.		
	<b>Note:</b> Options OR, OM, OS, OL, OA, OV, OC, and OK are available only at sites where CONTROL-O is installed.			
TR	RULE DEFINITION	Define rules.		
TP	POOL DEFINITION	Define pools.		
TV	VAULT DEFINITION	Define vaults.		
TI	INQ/UPD MEDIA DB	Display the Inquire/Update screen.		
TC	CHECK IN EXT VOL	Check in external volumes.		
<b>Note:</b> Options TR, TP, TV, TI, and TC are available only at sites where CONTROL-M/Tape is installed.				

### **IOA Version Information**

Enter **INFO** (or **I**) in the **OPTION** field of the IOA Primary Option menu to display the IOA Version Information window, as illustrated in Figure 5. This window lists the version and level of each INCONTROL product installed at the site, plus the CPU ID and current system date. The IOA Version Information window also identifies the unique IOA QNAME assigned to the site. For further information about the IOA

QNAME, see the IOA operational parameters step, the IOAPLEX parameters step, and the adding IOA structures to the CFRM step, all in the *INCONTROL for z/OS Installation Guide*. Press **Enter** or END (**PF03/PF15**) to exit the window and return to the IOA Primary Option menu.

Figure 5 IOA Version Information

	IOA PRIMARY OPTION MENU		(1)
OPTION ===>		USER	N06
IOA	CONTROL-D/V C		
	IOA VERSION INFORMATION	)       	- EFINITION ATISTICS TATUS TION LOG TION OPTS
	CONTROL-M/RESTART Version 6.1.00 Version 6.1.00 CONTROL-M/TAPE Version 6.1.00 Version 6.1.00 Version 6.1.00	)   )   )	STATUS CORDER
CONTROL-M & CTM	CONTROL-V Version 6.1.00 Version 6.1.00 Version 6.1.00		ape
2 JOB SCHEDU 3 ACTIVE ENV			EFINITION EFINITION
C CMEM DEFIN	DATE 19.08.01 CPUID 02078D 7060 IOA QNAME IOAR610		DEFINITION D MEDIA DB IN EXT VOL
COMMANDS: X - E	COMMANDS: X - EXIT, HELP, INFO OR CHOOSE A MENU OPTION 17.00.29		

### **Multi-Screen Control**

It is not necessary to return to the IOA Primary Option menu to move from one online facility to another.

To speed up transfer of control between screens of different facilities and to enable you to manage several online facilities at the same time, transfer control commands can be specified. Transfer commands take you directly from your current screen to the requested screen. Transfer commands can be used to reach any screen that can be accessed by the IOA Primary Option menu at your site.

Each transfer control command consists of an equal sign immediately followed by one of the options of the IOA Primary Option menu, which represents the target screen of the transfer. For example, from any screen, enter:

Table 21 IOA Transfer Control Commands

Command	Description
=5	to access the IOA Log screen
=4	to access the IOA Conditions/Resources screen
=1	to access the IOA Primary Option menu

If you use a transfer command to reach another screen, the state of the current screen remains unchanged when you return to it by another transfer command.

The INCONTROL administrator can globally deactivate any or all of the transfer commands.

# **Fast Exit from the IOA Online Facility**

To exit immediately from the IOA Online facility, type =X on the command line and press **Enter**.

In most cases, the =X command has the same effect as pressing END (**PF03/PF15**) in all open screens and then entering X (Exit) in the IOA Primary Option menu. Any window, such as the Exit Option window, that would be displayed when exiting an open screen is displayed when the =X command is entered.

However, when the =X command is entered while definition screens such as the Calendar Definition screen are open, changes to the open definition screens are cancelled. Changes currently in definition facility list screens, for example, changes to previously closed definition screens, are not cancelled. Those screens and all other open screens are treated as if END (**PF03/PF15**) has been entered.

#### NOTE -



The =X command is intentionally not supported on certain screens.

# **Screen Layout**

Most IOA screens are divided into four basic areas. The example used in this section is the IOA Log screen.

**Table 22** Basic IOA Screen Areas

Screen Area	Description
Screen Description and Message Line	This line at the top of the screen describes the purpose of the screen (in the example screen, "IOA Log"). A screen identifier may appear in the upper right corner (in the example screen, 5). This line is also used to display messages.
Screen Header and Command Area	This area is used for online commands, and, where applicable, headings of the screen data.
Data Area	On some screens, the data area can be scrolled. For more information, see "Scrolling Commands" on page 95.
Screen Bottom	This area of the screen usually contains a list of available commands or options (In the example screen, SHOW, GROUP, CATEGORY, and SHPF), or a brief explanation about screen usage. The current time is displayed in the lower right corner.

Figure 6 IOA Log Screen

FILTER:	IOA LOG(5)
COMMAND ===>	SCROLL===> CRSR
SHOW LIMIT ON ==>	DATE 291201 - 010102
DATE TIME ODATE USERID	CODE M E S S A G E
311201 184915 311201 K48	SUB13AI JOB K48RUN1 / OID=005W9 SUBMITTER STARTED PROCESSING JOB ON SYSTEM: OS35
311201 184915 311201 K48	SUB133I JOB K48RUN1 K48RUN /27255 OID=005W9
	SUBMITTED FROM LIBRARY (P) K48.LIB.JOB
311201 184918 311201 K48	SPY28GI JOB K48RUN1 K48RUN /27255 OID=005W9 TAPE
	DRIVE UNITS USED=00 00
311201 184918 311201 K48	SPY281I JOB K48RUN1 K48RUN /27255 OID=005W9 START
	01365.1849 STOP 01365.1849 CPU OMIN
	00.05SEC SRB OMIN 00.00SEC 0.00 4A0S35
311201 184918 311201 K48	SPY254I JOB K48RUN1 K48RUN /27255 OID=005W9
	SCANNED
311201 184918 311201 K48	SEL216W JOB K48RUN1 K48RUN /27255 OID=005W9
	UNEXPLAINED COND CODE 0015 STEP EXEC /
311201 184918 311201 K48	SEL214I JOB K48RUN1 K48RUN /27255 OID=005W9 RERUN
	NEEDED
311201 184918 311201 K48	SEL205I JOB K48RUN1 K48RUN /27255 OID=005W9 RERUN
	IN PROCESS USING MEM K48RUN1
311201 184918 311201 K48	SEL286I JOB K48RUN1 K48RUN /27255 OID=005W9
	WAITING FOR CONFIRMATION
CMDS: SHOW, GROUP, CATEGORY,	SHPF 08.57.11

# **Commands and PFKeys**

Commands are entered by typing a command in the **COMMAND** field and then pressing **Enter**, or by pressing a predefined PFKey, or a combination of both.

It is not necessary to enter the full command name; the shortest unique abbreviation of the command is sufficient. If the abbreviation is ambiguous, an appropriate message is displayed in the message area.

IOA commands are flexible; you can change command syntax or provide aliases (synonyms) to suit your site. If you want to add or change a command syntax, consult BMC Software Customer Support. The examples provided in this chapter exhibit the original command syntax supplied with this INCONTROL product.

PFKey command assignments can be site-customized. It is possible to assign PFKeys differently for each screen. To change PFKey command assignments, see your INCONTROL administrator.

Supplied PFKey definitions are consistent throughout most of the screens. For example: **PF08/PF20** is used to scroll down (forward) on all INCONTROL screens where scrolling is possible.

Table 23 Common PFKey Definitions (part 1 of 2)

PFKey	Description
PF01/PF13	HELP
PF02/PF14	SHOW (where applicable)
	<b>Note</b> : When the IOA Online facility is activated in ISPF mode (as an ISPF application), PF02/PF14 are usually assigned the ISPF SPLIT command. For more information, see "IOA Under ISPF" on page 101.
PF03/PF15	END – exit current screen and go back one level
PF04/PF16	RESET (where applicable)
PF05/PF17	FIND (where applicable)
<b>PF06/PF18</b> =6 – transfer to TSO screen/application or to UTILITIES so	
	<b>Note</b> : Disabled under ROSCOE/ETSO, CICS, VTAM, IMS/DC, IDMS/DC, COM-PLETE, and TSO cross memory option.
PF07/PF19	UP – scroll backward
PF08/PF20	DOWN – scroll forward
PF10/PF22	LEFT or PREV (where applicable)
PF11/PF23	RIGHT or NEXT (where applicable)

**Table 23 Common PFKey Definitions (part 2 of 2)** 

PFKey	Description
PF12	RETRIEVE – retrieves a sequence of commands and options entered by the user during the current session. These commands and options are displayed in reverse order on the command line of the current screen.
PF24	SHPF

To see the PFKey assignment of the screen with which you are working, type reserved command SHPF in the command line and press **Enter**. A window describing the current PFKey assignment appears on the screen. Press **Enter** again to close the window.

Figure 7 PFKey Assignment Window

FILTER:	- IOA LOG(5)
COMMAND ===>	SCROLL===> CRSR
SHOW LIMIT ON ==>	DATE 291201 - 010102
DATE TIME ODATE USERID CODE	M E S S A G E
311201 184915 311201 K48 SUB1:	3AI JOB K48RUN1 / OID=005W9 SUBMITTER STARTED
	PROCESSING JOB ON SYSTEM: OS35
311201 184915 311201 K48 SUB1:	33I JOB K48RUN1 K48RUN /27255 OID=005W9
	SUBMITTED FROM LIBRARY (P) K48.LIB.JOB
311201 184918 311201 K48 SPY28	BGI JOB K48RUN1 K48RUN /27255 OID=005W9 TAPE
	DRIVE UNITS USED=00 00
	31I JOB K48RUN1 K48RUN /27255 OID=005W9 START
+	
   FNTER ENTER	PF13 HFLP
PF01 HFIP	PF13 HELP PF14 SHOW
PF02 SHOW	PF15 FND
PF03 END	PF16 RESET
PF04 RESET	PF17 FIND
PF05 FIND	PF18 =6
PF06 =6	PF19 UP
PF07 UP	PF20 DOWN
PF08 DOWN	PF24 SHPF
PF12 RETRIEVE	
+	+

If you type text in the **COMMAND** field and press a PFKey, the text in the **COMMAND** field is treated as a subparameter of the command assigned to the PFKey.

Two additional key definitions are:

**Table 24 Additional Key Assignments** 

Key	Description
PA1	ABORT – forced exit  If you press PA1 while in AutoRefresh mode (described on
	page 101), AutoRefresh mode is canceled.
PA2	Under native TSO and ROSCOE, the first time you press this key, the screen is refreshed. The second consecutive time, a copy of the screen is sent to be printed, or to a file, using a PRTDBG DD statement. For terminal models supporting PA3, the PA3 key is defined in exactly the same way as PA2.
	When the IOA online facility is activated as an ISPF application, PA2 is controlled by ISPF, and only refreshes the screen. To print the screen, see "IOA Under ISPF" on page 101. Under other online environments, such as CICS and VTAM, PA2 serves as a refresh only. Usually one of the PA keys is assigned a local print function.

For information on changing IOA PFKey definitions, see the appendix in the *INCONTROL for z/OS Administrator Guide*, which deals with modifying IOA Online Facility Commands.

# **Scrolling Commands**

Scrolling conventions are very similar to the ISPF conventions of IBM. Two basic commands are used for scrolling:

**Table 25 Scrolling Commands** 

Command	PFKey	Description
UP	(PF07/PF19)	Scroll up (backward)
DOWN	(PF08/PF20)	Scroll down (forward)

The commands can be entered by typing the command in the **COMMAND** field or by pressing the predefined PFKey.

The scrolling amount is determined by the content of the **SCROLL** field in the right corner of the screen header. Valid scrolling amounts are:

Table 26 Scrolling Amounts in the SCROLL Field (part 1 of 2)

Scrolling Amount	Description
PAGE	Scroll a full page.
HALF	Scroll a half page.

Scrolling Amounts in the SCROLL Field (part 2 of 2) Table 26

Scrolling Amount	Description
CRSR	Scroll by cursor position. If the cursor is outside the data area, a full page is scrolled.
MAX	Scroll maximum available; for example, UP MAX scrolls to the top.

It is only necessary to type the first letter of the new amount in the SCROLL field in order to change the scrolling amount.

A scrolling amount other than that shown in the SCROLL field can be used by entering the amount directly after the scroll command itself, or by entering the scroll amount in the COMMAND field and pressing the appropriate scrolling PFKey. The scrolling amount in the SCROLL field remains unchanged.

### **Example**

If PAGE is the value in the SCROLL field, to scroll to the bottom, type M (MAX) in the **COMMAND** field and press **PF08** (DOWN).

### **LOCATE Command**

The LOCATE command, and its abbreviation, L, can be used to search for items in the NAME field in all "directory type" screens that contain scrollable data, such as the Calendar List screen. The syntax of the command is

LOCATE string

where *string* is the search string. Apostrophes ('single quotes') or quotation marks ("double quotes") are not required.

The search proceeds from the top of the list to the first item in the list that starts with the specified string. The cursor is positioned on the **OPTION** field at the beginning of the line containing the string, if found, or on the OPTION field of the alphabetically closest preceding value if the specified value is not found.

### **FIND Command**

The FIND command, and its abbreviation, F, can be used in all screens that contain scrollable data to find and display the next occurrence of a character string. The syntax of the command is

FIND string [fromcol] [tocol] [PREV]

#### where:

- string is the search string Mandatory.
- fromcol is the first column in the search range Optional.
- tocol is the last column in the search range Optional.
- PREV is the indicator that the search must move backward, instead of forward, from the current cursor position Optional.

#### **General Rules**

If the string contains blanks, enclose the string with apostrophes ('single quotes') or quotation marks ("double quotes"). For example:

#### FIND 'WAIT SCHEDULE'

The column range searched can be limited by entering *fromcol* or *tocol* values, or by entering both *fromcol* and *tocol* values.

The search for the string proceeds from the current cursor position forward, or backward if PREV is entered. If the string is found, the cursor is positioned at the start of the string.

To repeat the find, to the next or previous occurrence of the string, press PF05/PF17.





The following situations outline where the FIND command can, or should, be further modified to enhance its functionality.

- Some screens enable the user to limit the number of lines searched by a FIND command. This is discussed in the relevant screen descriptions.
- In some screens, the FIND command does not detect information that is to the right or left of the information displayed in the monitor. To ensure detection of the desired string, the screen must be displayed in wraparound mode, when available, before executing the FIND command.

### **Text String Searches**

The FIND command can also be used to search for text strings, in which case the command will find all instances of the string, regardless of whether the characters within the string are lowercase, uppercase, or mixed case. To search for a text string, include the letter *T* immediately before a quoted string.

For example,

#### FIND T'WAIT SCHEDULE'

will find WAIT SCHEDULE, and it will also find wait schedule, and Wait Schedule, and any other case variant.

Text string searches are the default. If your system default is for text strings, You do not need to include the *T* if you perform a text string search. Your INCONTROL administrator can change the default to character string. In this case you do not need to include the *C* if you perform a character string search.

### **Character String Searches**

The FIND command can be used to search for character strings, in which case the command will find all instances of the string, but only where the string contains characters that match the case specified. To search for a character string, include the letter *C* immediately before a quoted string.

For example,

#### FIND C'WAIT SCHEDULE'

will find WAIT SCHEDULE, but it will not find wait schedule, or Wait Schedule, or any other case variant.

### CANCEL and RESET Commands

CANCEL and RESET commands are entered in the COMMAND field.

The CANCEL command cancels changes made in a definition screen, such as the IOA Calendar Definition screen, and exits the screen.

The RESET command (PF04/PF16) cancels Edit environment options specified in a definition screen. It does not cancel changes already made and it does not exit the screen or cancel Edit environment mode. For more information about the Edit environment, see Appendix A, "The CONTROL-M Application Program Interface (CTMAPI)."

The RESET command (PF04/PF16) can also be used in most windows, for example, the Show Screen Filter window, to cancel changes and close the window.

# **Online Help**

The following types of online help are available for INCONTROL screens:

### **Screen help**

Provides information about the entire screen. This help is available on all INCONTROL screens and is accessed by pressing the HELP key (**PF01/PF13**) while the cursor is positioned on the **COMMAND** field in the screen.

# **Line-Sensitive Help**

Provides information about the fields on a particular line on a screen. This help is available on several INCONTROL screens. It is accessed by pressing the HELP key (PF01/PF13) while the cursor is positioned on the desired line of the screen.

If line-sensitive help is not supported in a screen, pressing the HELP key (**PF01/PF13**) from anywhere in the screen displays the beginning of the Help panel.

Figure 8 IOA Help Screen

```
COMMAND ===>
                                                          SCROLL===> CRSR
 Calendar List Screen
 The Calendar List screen displays a list of calendars (members) in the
 specified library. This screen can be entered directly from the entry
 panel or upon exiting the Year List screen.
 By default, only calendar names are listed in the screen. However, if
 the default has been modified at time of installation, statistical
  information is displayed for each calendar name.
 Use the scrolling PFKeys to scroll forward (PF08/PF20) and backward
  (PF07/PF19) on the Calendar List.
  To return to the entry panel, press END (PF03/PF15).
 Options of the Calendar List Screen
 To request one of the following options, specify the option in the OPT
ENTER END OR PF03/PF15 TO EXIT THE HELP SCREEN
```

Help can be scrolled using standard scrolling conventions.

To return to the original screen, use the END command (PF03/PF15).

The Help member name appears on the right in the Help screen header. Members containing the Help descriptions can be found in the IOA MSG library.

### **AutoRefresh Mode**

Certain INCONTROL screens, as noted in this chapter where appropriate, support AutoRefresh mode. A screen display in AutoRefresh mode is automatically updated periodically with the most current data.

AutoRefresh mode can only be activated under native TSO or under ISPF. AutoRefresh mode is activated by the AUTO command. The format of the command is

AUTO n

where *n* is any number of seconds from 1 through 99.

The screen is updated when the AUTO command is issued, and then periodically updated according to the interval (in seconds) specified in the AUTO command. A counter at the top of the screen displays the number of times the screen has been refreshed.

#### - NOTE



Issuance of the AUTO command may be controlled via the IOA security interface. See the *INCONTROL for z/OS Security Guide* for further details.

### **Example**

The AUTO 5 command refreshes the screen every 5 seconds.

### **Cancelling AutoRefresh Mode**

Under native TSO, the recommended method of cancelling AutoRefresh mode is as follows:

■ For short interval values – Press **Enter**. Whenever **Enter** is pressed, or a command is issued, AutoRefresh mode is automatically cancelled at the end of the current interval.

■ For long interval values – Press Attn (PA1) once.

Under ISPF, press Attn (PA1) or Esc once to cancel AutoRefresh mode.

### **IOA Under ISPF**

The IOA Online facility can be activated as an ISPF application. As such, it can work in ISPF split screen mode like any other ISPF application.

### WARNING



Multiple calls to the IOA ISPF interface can be performed in ISPF split screen mode as long as all invocations are for the same IOA environment. Otherwise, the results may be unpredictable.

The command line of the IOA Online facility is controlled by IOA. It is not possible to enter ISPF commands in an IOA screen. Two ISPF commands must be defined to PFKeys:

Table 27 ISPF Commands that must be defined for PFKeys

Command	PFkey
SPLIT	(usually PF02/PF14)
SWAP	(usually PF09/PF21)

The rest of the PFKeys are controlled by IOA PFKey definitions, which are in the IOA PARM library.

It is possible to assign TSO/ISPF commands such as PRINT to PFKeys, or to change PFKey definitions by performing the following steps:

- **1** Exit from IOA and ISPF to the READY prompt.
- **2** Type the following command and press **Enter**:

#### ISPSTART PANEL(ISR@PRIM) NEWAPPL(CTM)

This command brings you to ISPF.

- **3** Type the KEYS command and press **Enter**. A set of key definitions is displayed.
- **4** Modify the key definitions as desired and exit from ISPF.



#### — NOTE –

ISPF KEY definitions for the following ISPF commands take precedence over IOA PFKey definitions: SPLIT, SWAP, KEYS, PRINT, PFSHOW. For example, if **PF02** is defined as SPLIT in ISPF, an IOA definition for **PF02** is ignored in online screens.

For all other ISPF commands, such as UP or DOWN, the key definitions in ISPF are ignored and the PFKey is interpreted according to the definition in the IOA Online facility.

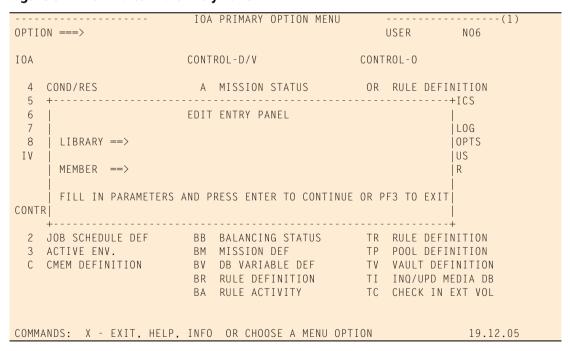
Under ISPF, IOA Option 6 activates the Online Utilities panel, which is described in "IOA Online Utilities Menu" on page 323. For more information about these utilities, see the *INCONTROL* for *z/OS Utilities Guide*.

For more information on changing IOA PFKey definitions, see the appendix in the *INCONTROL for z/OS Administrator Guide* that deals with modifying IOA Online Facility Commands.

### **IOA Editor**

The IOA Editor enables you to edit members of a partitioned data set (PDS) using an editor similar to the ISPF editor. Enter **EDMEM** in the command line of any screen to display the Edit Entry Panel window, as shown in Figure 9.

Figure 9 IOA Editor Edit Entry Panel



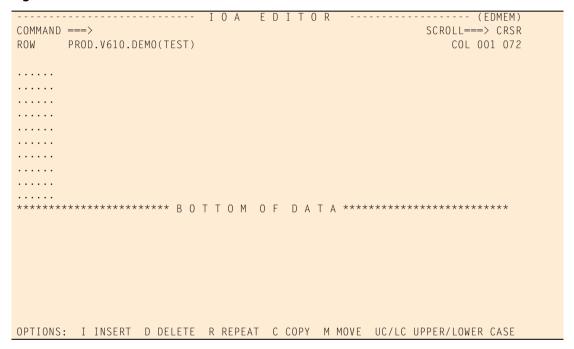
To create a new member or edit an existing member, fill in the LIBRARY and MEMBER parameters and press **Enter**. The IOA Editor screen is opened for editing, as shown in Figure 10.

### _ NOTE _



If the member already exists in the specified library, the member is displayed for editing in the IOA Editor. Similarly, if you accessed the IOA Editor screen from line option J in either screen 2 or screen 3, the member in the library referred to in the schedule definition member will be displayed for editing.

Figure 10 IOA Editor



# **IOA Editor PFKey Functions**

While working within the IOA Editor, PFKeys perform the functions shown in Table 28:

Table 28 PFKey Functions Within the IOA Editor Screen (part 1 of 2)

PFKeys	Description
PF01/PF13	Activates online help.
PF02/PF14	Saves the current member.
PF03/PF15	Terminates the editing session. If the edited member has been changed the member will be saved automatically.
PF04/PF16	Cancels the editing session without saving changes.
PF05/PF17	Invokes the Find facility.
PF07/PF19	Scrolls forward.

Table 28 PFKey Functions Within the IOA Editor Screen (part 2 of 2)

PFKeys	Description
PF08/PF20	Scrolls backward.
PF10/PF22	Scrolls left.
PF11/PF23	Scrolls right.

### **Commands of the IOA Editor Screen**

Table 29 describes editing commands that can be executed by entering the command in the COMMAND line.

Table 29 IOA Editor Command Line Commands

Command	Description
SAVE	Saves all new data without terminating the edit session.
CANCEL	Terminates the edit session without saving new data.
COPY	Enables you to import a member from a specific library.

Table 30 describes editing commands that can be executed by entering the command in the left-most position of the applicable row.

**Table 30 IOA Editor Row Commands (part 1 of 2)** 

Command	Description
I	Inserts a new line below the current line.  To insert more than one line for new data, enter <b>Inn</b> , where <i>nn</i> indicates the number of new lines to be inserted below the current line.
D	Deletes the current line.  To delete more than one line, enter <b>Dnn</b> , where <i>nn</i> indicates the number of lines to be deleted below the current line.  You can delete a block of lines by typing <b>DD</b> at the beginning of the first line of the block, and then entering <b>DD</b> at the beginning of the last line of the block.
R	Repeats the current line.  To repeat a single line one or more times, enter <b>R</b> nn, where nn indicates the number of times the current line is to be repeated.  You can repeat a block of lines by typing <b>RR</b> at the beginning of the first line of the block, and then entering <b>RR</b> at the beginning of the last line of the block.
С	Identifies the source line for a copy operation.  To copy more than a single line, enter <i>Cnn</i> , where <i>nn</i> indicates the number of lines to be copied.  You can also copy a block of lines by typing <i>CC</i> at the beginning of the first line of the block, and then entering <i>CC</i> at the beginning of the last line of the block.

Table 30 IOA Editor Row Commands (part 2 of 2)

Command	Description
M	Identifies the source line for a move operation.  To move more than a single line, enter <b>M</b> nn, where nn indicates the number of lines to be moved.  You can also move a block of lines by typing <b>MM</b> at the beginning of the first line of the block, and then entering <b>MM</b> at the beginning of the last line of the block.
A	Identifies the destination of a copy or move operation.  When a line or block of lines has been selected for copying or moving, enter A at the point after which the copied lines are to be inserted.
В	Identifies the destination of a copy or move operation.  When a line or block of lines has been selected for copying or moving, enter <b>B</b> at the point before which the moved lines are to be inserted.
LC	Changes text in a line from uppercase to lowercase.  To change text in more than a single line to lowercase, enter <b>LC</b> <i>nn</i> , where <i>nn</i> indicates the number of lines to be changed to lowercase.
UC	Changes text in a line from lowercase to uppercase.  To change text in more than a single line to uppercase, enter <b>UC</b> <i>nn</i> , where <i>nn</i> indicates the number of lines to be changed to uppercase.

# **IOA SET Command Panel**

The IOA SET Command Panel enables you to set and stop TRACE levels, choose the language that is used in online screens and to set a dollar sign representation that will be used in online screens for system variables of type %%\$VAR. Enter SET in the command line of any screen to display the SET Command Panel window, as shown in Figure 11.

#### Figure 11 IOA SET Command Panel

```
SET Command Panel

TRACE level , ON (Trace level 001-512, ON or OFF)

LANGUAGE ENG - English
FRA - French
GER - German
JPN - Japanese

SPECIAL CHARACTERS: Dollar Current representation is "$"(X'5B')

FILL IN PARAMETERS AND PRESS ENTER TO CONTINUE OR PF3 TO EXIT
```

The process of setting TRACE levels and turning off a particular TRACE, and the process of setting language preferences for online screens and messages, begins in the SET Command Panel.

# **Using the SET Command Panel to set and end TRACE Levels**

Setting the TRACE level can help you monitor certain IOA Online facility and INCONTROL functions, such as security checks.

The following steps explain how to set or turn off a TRACE level:

- 1 Type a TRACE level number, from 1 through 256, in the TRACE level field of the SET Command Panel.
- 2 In the (Trace level 1-256, ON or OFF) field, type ON to set a TRACE level, or OFF to turn off a TRACE level.
- 3 Press Enter to confirm the setting, in which case the following message is displayed:

```
CTMA2AI TRACE LEVEL nnn WAS SET xxx
```

#### where

- *nnn* is the TRACE level number
- xxx indicates whether the TRACE level was set ON or turned OFF





TRACE level settings take effect immediately.

# Using the SET Command Panel to set a dollar sign representation

Setting the dollar sign representation influences how System variables are shown in online screens.

The following steps explain how to set the dollar sign representation:

- 1 Type a \$ character in the Dollar field using your keyboard.
- 2 Press Enter to confirm the setting, in which case the following message is displayed:

CTMA2DI THE NEW DOLLAR REPRESENTATION IS "c"(X'yy')

#### where

- $\blacksquare$  c is \$ character you set
- yy is the EBCDIC hexadecimal code for \$

#### NOTE



There are differences in the EBCDIC hexadecimal code for the \$ (dollar sign) character on keyboards that have been adapted to show local or national symbols.

# **Using the SET Command Panel to set Language preferences**

Setting the LANGUAGE influences the online screens and messages in subsequent sessions.

The following steps explain how to set language preferences:

- 1 In the LANGUAGE field, type one of the following sets of characters to select a language preference:
  - ENG, to set English as the preferred language
  - FRA, to set French as the preferred language
  - GER, to set German as the preferred language
  - JPN, to set Japanese as the preferred language

**2** Press **Enter** to confirm the setting, in which case the following message is displayed:

CTMA27I THE NEW LANGUAGE WILL BE USED FROM THE NEXT LOGON TO IOA





Language preference settings do not take effect until your next logon to the system.

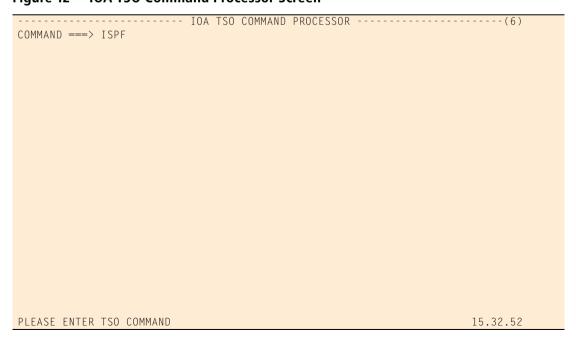
### **IOA TSO Command Processor Screen**

The IOA TSO Command Processor screen can be entered only when the IOA Online facility is activated as a TSO application. It cannot be entered when the IOA Online facility is activated as an ISPF application or activated under a non-TSO environment.

The TSO screen enables activation of any TSO command without exiting the IOA Online facility. For example, a typical program activated under the TSO screen is ISPF. Therefore all ISPF/PDF facilities and functions, such as editing a member or scanning job output, can be activated while you are working under the IOA Online facility.

To activate a TSO command, type the command in the **COMMAND** field and press **Enter**.

Figure 12 IOA TSO Command Processor Screen







CLISTs cannot be activated from the TSO screen. To activate a CLIST, first activate ISPF and then execute the CLIST under ISPF.

TSO commands can also be activated directly from any IOA online screen by typing TSO in the **COMMAND** field.

# Transfer of Control Between the TSO Application and the IOA Online Facility

You can return to the IOA Online facility from the TSO application by simply exiting the TSO application in a normal manner. However, this method can be time consuming and inconvenient if an ISPF application or a similar TSO application is activated.

If the TSO application can issue a TSO command, it is possible to transfer control to the IOA Online facility, and vice versa, without exiting the TSO application.

While working under the TSO application, for example, under ISPF, issue the command:

TSO CTMTTRA  $\{n \mid =n\}$ 

where n is the online screen number.

The requested screen is displayed as it was when you transferred from it.

To return to the TSO application, use the =6 command (**PF06/PF18**). The application remains in the same state as when you transferred from it.

It is recommended that you simplify transfer between screens by permanently assigning one of your PFKeys under ISPF (or SDSF, and so on) to the command TSO CTMTTRA. Once this key assignment is made, you no longer need to type the full transfer command. Instead, you merely type the IOA option number or code in the **COMMAND** field and press the assigned PFKey. You are transferred to the desired screen.

#### - NOTE



You must activate ISPF under the IOA Online facility if you want to use the control transfer feature.

## **Scheduling Definition Facility**

The CONTROL-M Scheduling Definition facility enables you to create, view, or modify job scheduling definitions for the jobs in your environment. A job scheduling definition consists of parameters that correspond to the decisions and actions of the operator when handling the scheduling, submission, and post-processing of a job.

The job scheduling definition for a job needs to be defined only once. Once defined, the definition is saved and used as necessary for managing job processing. Job scheduling definitions can be modified or deleted as required.

Job scheduling definitions are stored in members called scheduling tables. Any number of scheduling tables can be defined, and each scheduling table can contain any number of job scheduling definitions.

In many production environments, related applications are scheduled together as a group. In these cases, it is common to define all such related applications in a single scheduling table, and to schedule all the jobs in the table together, as a group.

Scheduling tables (members) are stored in scheduling libraries (partitioned data sets). You can define any number of scheduling libraries. The number of scheduling tables in a library, the number of job scheduling definitions in a scheduling table, and the size of each job scheduling definition, are all calculated dynamically.

### **NOTE**



The CONTROL-M Scheduling Definition facility does not support members that have been compressed using the ISPF PACK option.

## **Accessing the Scheduling Definition Facility**

The Scheduling Definition facility contains the following screens:

**Table 31 Scheduling Definition Facility Screens** 

Screen	Definition
Scheduling Facility entry panel	Enables specification of parameters that determine which screen is displayed.
Table List screen	Displays the list of tables (members) in the specified scheduling library.
Job List screen	Displays the list of jobs (job scheduling definitions) in the selected table.
Job Scheduling Definition screen (Screen 2)	Displays the parameters of the selected job scheduling definition or Group Entity. This is the main screen of the facility.

To enter the Scheduling Definition facility, select option 2 on the IOA Primary Option menu. The Scheduling Definition Facility entry panel is displayed.

## **Handling of Job Groups**

A group of jobs whose processing (for example, scheduling, submission, and post processing) is handled as a group, is defined in its own scheduling table, called a Group scheduling table. This table must be created with G (Group) inserted in the **TYPE** field of the Scheduling Definition facility entry panel.

At time of creation of a Group scheduling table, a special scheduling definition, called a Group Entity, is created. The Group Entity is used to define job processing criteria for the group as a whole. These include:

**Table 32 Scheduling Criteria** 

Criterion	Description
Basic Scheduling Criteria	Any number of sets of basic scheduling criteria can be specified in the Group Entity. At least one of these sets must be satisfied before the group, or any job in the group, can be scheduled.
	Each set of basic scheduling criteria in the Group Entity is assigned a unique name called a Schedule Tag.
	These Schedule Tag names can be entered in job scheduling definitions in the table. When a set of basic scheduling criteria in the Group Entity is satisfied, job scheduling definitions that specify the corresponding Schedule Tag are scheduled that day.
	Job scheduling definitions can also contain their own basic scheduling criteria, and be scheduled according to those criteria, provided that the group itself can be scheduled.
Runtime Scheduling Criteria	Before any job in a group can be considered for submission, all group runtime scheduling criteria specified in the Group Entity must be satisfied. Once these are satisfied, a job is submitted only if its own specified runtime scheduling criteria are satisfied.
Post-processing Actions	Post-processing actions can be defined for the group, in the Group Entity. These are performed once the group has finished processing (that is, all jobs in the group have terminated).
	These actions can be made conditional upon whether all submitted jobs in the Group scheduling table ended OK, or whether at least one job did not end OK.

The Group Entity also contains a field (ADJUST CONDITIONS) that enables job dependencies based on prerequisite conditions to apply only if predecessor jobs in the group are scheduled.

CONTROL-M internally tracks each job group and the jobs in the group. Each order of each group of jobs is identified as a unit.

The status of each job group that has been ordered can be viewed using option G (Group) of the Job Status screen (Active Environment screen).

### NOTE



When the IN conditions of a Group entity are satisfied (for example, they have been added to the IOA Conditions file), the jobs in the group begin execution, assuming that their other runtime criteria are satisfied.

By default, if jobs in a group have already begun execution and an IN condition for the job group is deleted from the IOA Conditions file, this change does not affect the processing of the jobs in the group; the jobs continue execution as if all the IN conditions were still satisfied. This default is overridden if the GRPRECHK parameter in the CTMPARM member in the IOA PARM library is set to Yes, in which case IN conditions in the Group entity are checked before each job is selected.

## **Creating Tables**

Tables can be created in any of the following ways:

- by typing the new table name in the entry panel and pressing Enter
   The name of a new job scheduling definition for the new table can also be entered.
- by using the SELECT command to choose the new table name in the Table List screen and pressing Enter

The SELECT command is described in "The SELECT Command" on page 122.

When you enter a create table request using either of the above methods, a skeletal job scheduling definition is displayed in the Job Scheduling Definition screen if a Job scheduling table is being created. Fill in and save this job scheduling definition. The table is created and the job scheduling definition is the first and only job scheduling definition in the Job list of the table. As additional job scheduling definitions are created in the table (described below), they are added to the Job list.

When you enter a create table request using either of the above methods, a skeletal Group Entity scheduling definition is displayed in the Scheduling Definition screen if a Group scheduling table is being created. Fill in and save this Group Entity.

#### NOTE



The PGRPEMPT profile variable controls whether empty group tables (without a job scheduling definition) may be created.

Valid values are:

- Y (Yes) empty group tables may be created
- N (No) empty group tables may not be created. All group tables must be created with at least one job scheduling definition. Default.

When PGRPEMPTY=Y, the table is created and only the Group entity is in the Job List of the table. Adding a job scheduling definition to this table is described in "Creating Job Scheduling Definitions" on page 113.

When PGRPEMPTY=N, a skeletal job scheduling definition is displayed in the Job Scheduling Definition screen. Fill in and save this job scheduling definition. The table is created and the Group Entity and the job scheduling definition are in the Job list of the table. As additional job scheduling definitions are created in the table (described below), they are added to the Job list.

#### NOTE



Upon exiting the Job List screen, if changes were made in at least one job scheduling definition, an Exit Option window is displayed. One field of the window displays the table name. This value can be changed to a new table name. This creates a new table in which the job scheduling definitions are saved.

Under ISPF, tables can also be created using the M5 online utility. This method is described in "M5: Quick Schedule Definition" on page 360, and is not included in this discussion.

## **Creating Job Scheduling Definitions**

Job scheduling definitions can be created using two basic methods:

- A skeletal job scheduling definition can be created by typing the name of a new job scheduling definition in the entry panel. The table specified in the entry panel can be either a new table if PGRPEMPT=N or an existing table if PGRPEMPT=Y. In this case, virtually all fields of the job scheduling definition are empty.
- A copy of an existing job scheduling definition can be created using the INSERT option in the Job List screen, described in "Options of the Job List Screen". In this case, most fields of the new job scheduling definition have the same values as the fields in the copied job scheduling definition.

#### NOTE -



Under ISPF, job scheduling definitions can also be created using the M5 online utility. This method is described in "M5: Quick Schedule Definition" on page 360, and is not included in this discussion.

## **Performing Operations on Tables and Jobs**

Many operations can be performed on tables and on the job scheduling definitions in them. These operations are performed through commands and options in the various screens of the Scheduling Definition facility.

Some of the major operations possible within the facility are described in the following pages. Options and commands that have not yet been explained are explained in detail following the summary.

## Accessing (Editing or Browsing) a Table and its Jobs

A table (that is, the job scheduling definitions in the table) can be browsed or edited.

When browsed, the table cannot be modified or updated. When the table is edited, new job scheduling definitions can be added and existing job scheduling definitions can be modified or deleted.

Browsing, however, has advantages:

- Access and exit are quicker than in editing.
- Job lists and job scheduling definitions that are in use by another user can be viewed.
- Access for browsing might be granted, even though access for editing might be denied due to site security requirements.

#### NOTE



Users who request Edit mode, but do not have edit authority, can be automatically forced to the Browse mode by the IOA Administrator. See the IOA Administrators Guide, Chapter 10, exit IOA032 for further information.

To browse a table (and its job list and job scheduling definitions) use the BROWSE option in the Table List screen.

Entering the table name in the entry panel or using the SELECT option in the Table List screen provides edit access.

Depending on user profile definitions, if the table requested for editing is in use, either access is granted in Browse mode or access is not granted.

## Accessing the JCL of a Job

When IOA is activated under ISPF, the member containing the JCL of a job can be accessed by the JCL command in the Job List screen. Whether the member can be modified and updated depends on whether the Job List screen was accessed in Browse or Edit mode.

## Copying a Job to Another Table

Jobs can be copied from one table to another by the COPY option in the Job List screen. For more information, see "Options of the Job List Screen" on page 367.

## **Deleting a Table or a Job**

Unneeded jobs can be deleted using the DELETE option in the Job List screen. For more information, see "Options of the Job List Screen" on page 125. Unneeded tables can be deleted by the DELETE option in the Table List screen. For more information, see "Deleting Tables" on page 158.

#### - NOTE



In order to delete the last (or only) job scheduling definition from a group table, profile variable PGRPEMPT must be set to **Y**.

## **Displaying Jobflow in Graphic Format**

The job flow of jobs in a table can be displayed in graphic format by the GRAPHIC FLOW option in the Table List screen. For more information, see "Displaying Graphic Jobflow" on page 162.

## **Displaying Job Statistics**

The statistics for a job can be displayed by performing any of the following:

- Typing **S** (STAT) next to the job name in the Active Environment screen.
- Typing **T** (JOBSTAT) next to the job name in the Job List screen.
- Typing the primary command JOBSTAT in the Job Scheduling Definition screen (or the Active Environment screen).

### **Manually Scheduling Jobs**

Manually ordering a job results in the job being scheduled only if its basic scheduling criteria are satisfied. Manually forcing a job results in its being scheduled even if its basic scheduling criteria are not satisfied.

- To manually order all the jobs in a table, type **O** (**ORDER**) for the table in the Table List screen. Multiple tables can be ordered.
- To manually force all the jobs in a table, type **F** (**FORCE**) for the table in the Table List screen. Multiple tables can be forced.
- To manually order specific jobs in a table, type **O** (**ORDER**) for the jobs in the Job List screen.
- To manually force specific jobs in a table, type **F** (**FORCE**) for the jobs in the Job List screen.

For more information, see "Ordering (Scheduling) Jobs" on page 150.

## Displaying the Schedule Plan of a Job

The schedule of a job for a specified period of time, based on the basic scheduling criteria of the job, can be displayed in calendar format by PLAN option in the Job List screen. For more information, see "Displaying a Job Scheduling Plan" on page 164.

## Simulating the action of the CONTROL-M submission mechanism for a job

A job's JCL member can be tested for proper auto-edit resolution by typing % (AutoEdit simulation) next to the job name in the Job List screen, or next to the job name in the Active Environment screen (after the job is ordered). For more information, see the % option in Table 35 and Table 58.

## **Saving Modifications**

All changes made to a table and its job scheduling definitions are kept in memory until the table is exited. Upon exiting the table, you can choose to save or cancel the changes. For more information, see "Exiting the Scheduling Definition Facility" on page 148.

## **Entry Panel**

The entry panel is displayed upon entering the Scheduling Definition facility (Option 2 in the IOA Primary Option menu).

Figure 13 CONTROL-M Scheduling Definition Facility - Entry Panel

```
------ CONTROL-M SCHEDULING DEFINITION FACILITY – ENTRY PANEL ------(2)
 COMMAND ===>
 SPECIFY LIBRARY, SCHEDULING TABLE, JOB
    LIBRARY ===> CTM.PROD.SCHEDULE
    TABLE ===>
                                    (Blank for table selection list)
    J0B
                                    (Blank for job selection list)
    TYPE OF TABLE
                                    ( J Job - default
                                      G Group - for new tables only)
 SHOW JOB DOCUMENTATION ===> N
                                    (Y/N)
 AUTO-SAVE DOCUMENTATION ===> N
                                    (Y/N)
USE THE COMMAND SHPF TO SEE PFK ASSIGNMENT
                                                                      23.00.04
```

To open the desired display, fill in Entry Panel fields **LIBRARY**, **TABLE**, and **JOB** as described below. Type **J** (scheduling table for individual jobs) or **G** (scheduling table for jobs handled as a group) for TYPE OF TABLE if you are creating a new scheduling table. If you are not creating a new table, the **TYPE OF TABLE** field is ignored and all types of tables are displayed.

Type **Y** (Yes) or **N** (No) in the **SHOW JOB DOCUMENTATION** field to determine whether job documentation lines appear when the Job Scheduling Definition screen is displayed. Type **Y** (Yes) or **N** (No) in the **AUTO-SAVE DOCUMENTATION** field to determine whether changes made to documentation are automatically saved when updating the job scheduling definition.

- To display the list of tables in a library, do the following:
  - 1. Type the library name.
  - 2. Either leave the table name blank, or type part of a table name together with mask characters (* and ?).
  - 3. Press Enter.
- To display the list of jobs of a specific table, do the following:
  - Type the library name.
  - 2. Type the table name.
  - 3. Press Enter.

If the table does not exist, the screen for defining a new job in the table is displayed.

- To display the details of a specific job (Job Scheduling Definition screen), do the following:
  - 1. Type the library name.
  - 2. Type the table name.
  - 3. Type the job name.
  - 4. Press Enter.

If the table does not exist, or the job for the specified table does not exist, the screen for defining a new job in the table is displayed.

### NOTE



If you enter the screen for defining a new job and want to leave the screen without defining a job, use the **CANCEL** command.

- To display the Search Window (described below), do the following:
  - 1. Type the library name.
  - 2. Type the job name.
  - 3. Either leave the table name blank, or type part of a table name together with mask characters (* and ?).
  - 4. Press Enter.
- To create a new table, do the following:
  - 1. Type a new table name.
  - 2. Type the table type.
  - 3. Press Enter.

The Job Scheduling Definition screen, for defining the first job in the new table, is displayed.

## **Search Window**

The Search window enables you to search for the specified job in tables in the specified library. Tables in which the job has been found are then displayed in the Table List screen.

Figure 14 CONTROL-M Scheduling Definition Facility - Entry Panel Search Window

```
------- CONTROL-M SCHEDULING DEFINITION FACILITY - ENTRY PANEL ------(2)
COMMAND ===>
SPECIFY LIBRARY, SCHEDULING TABLE, JOB
   LIBRARY ===> CTM.PROD.SCHEDULE
   TABLE ===>
                                   (Blank for table selection list)
          ===> CTMCLRES
                                   (Blank for job selection list)
   TYPE OF TABLE
                        ===>
                                   PLEASE SELECT ONE OF THE FOLLOWING:
                                      - STOP SEARCH IMMEDIATELY
                                      - ASK AGAIN AFTER 000010 TABLES
SHOW JOB DOCUMENTATION ===> N|
                                   3 - UNCONDITIONAL SEARCH
AUTO-SAVE DOCUMENTATION ===> N |
                                   NUMBER OF TABLES IN LIBRARY: 000015
                                   NUMBER OF SEARCHED TABLES:
                                   NUMBER OF SELECTED TABLES:
USE THE COMMAND SHPF TO SEE PFK ASSIGNMENT
```

To close the Search Window without performing any action, press END (PF03/PF15).



– NOTE –

If you use the selection list fields, their values are not erased until you exit the entry panel by pressing END (PF03/PF15).

To perform a search, select one of the following choices and press **Enter**:

```
3 - UNCONDITIONAL SEARCH
```

Searches all tables in the specified library.

The search continues uninterrupted unless and until you select Option 1 (Stop Search Immediately).

```
2 - ASK AGAIN AFTER number TABLES
```

Searches the specified number of tables in the specified library, and then pauses. The search number can be modified. Default: 10.

- Continue the search by pressing Enter.
- Stop the search by selecting option 1 (Stop Search Immediately).

If any tables are found, the Table List is displayed listing those tables.

During the search, the following information is displayed at the bottom of the window:

- Number of tables in library. Lists the total number of tables in the specified library.
- Number of searched tables. Lists the cumulative number of tables searched. For example, if you perform three searches with a specified number of 10, the figure displayed is 30.
- Number of selected tables. Lists the cumulative number of tables selected that contain the job being searched.

If any tables are selected during the search, the Table List is displayed listing those tables. If no tables are selected, the Search Window is closed and a message is displayed.

## **Table List screen**

The Table List screen displays a list of scheduling tables (members) in the specified library. This screen can be entered directly from the entry panel or upon exiting the Job List screen.

By default, only table names are listed in the screen. However, if the default has been modified at time of installation, statistical information is displayed for each table name, as shown in the following screen example.

LIST OF TABLES IN CTM.PROD.SCHEDULE COMMAND ===> SCROLL ===> CRSR OPT NAME ----- VV.MM CREATED CHANGED SI7F INIT MOD ID 01.00 01/02/14 01/06/12 00:50 0 001 ASMBTR1 01.00 01/02/14 01/06/12 00:50 41 41

BACKUP 01.00 01/02/14 01/06/12 00:50 5 5

CICSJOBS 01.00 01/02/14 01/06/12 00:50 5 5

CICSSPROD 01.00 01/02/14 01/06/12 00:50 70 70

CICSTEST 01.00 01/02/14 01/06/12 00:50 41 41

CICSUPT 01.00 01/02/14 01/06/12 00:50 5 5

CLIENTS 01.00 01/02/14 01/06/12 00:50 5 5

CLIENTS 01.00 01/02/14 01/06/12 00:50 5 5

CHIENTS 01.00 01/02/14 01/06/12 00:50 5 5

CLIENTS 01.00 01/02/14 01/06/12 00:50 5 5

MAINDAY 01.00 01/02/14 01/06/12 00:50 5 5

MAINDAY 01.00 01/02/14 01/06/12 00:50 5 5

MAINDAY 01.00 01/02/14 01/06/12 00:50 5 5

MAINT 01.00 01/02/14 01/06/12 00:50 5 5

ONSPOOL 01.00 01/02/14 01/06/12 00:50 9 9

ONSPOOLT 01.00 01/02/14 01/06/12 00:50 9 9

ONSPOOLT 01.00 01/02/14 01/06/12 00:50 9 9

ONSPOOLT 01.00 01/02/14 01/06/12 00:50 9 9

ONS S SELECT 0 OPPER ADABAS 0 001 0 001 5 0 S07 9 0 S07 14 0 S07 5 0 S07 0 001 41 5 9 0 001 0 001 0 S07 0 S07 14 0 S07 0 S07 0 001 0 001 0 S07 0 S07 0 S07 0 S07 OPTIONS S SELECT O ORDER F FORCE G GRAPHIC FLOW B BROWSE D DELETE 15.38.37

Figure 15 CONTROL-M Scheduling Definition Facility Table List Screen

- To scroll down the Table list, press **PF08/PF20**. To scroll up the Table list, press **PF07/PF19**.
- To return to the entry panel, press END (**PF03/PF15**).

## **Options of the Table List Screen**

To request one of the following options, type the option in the **OPT** field to the left of the table names and press **Enter**.

Table 33 Options of the Table List Screen (part 1 of 2)

Option	Description
S (SELECT)	Display the list of jobs in the table for any purpose, including editing and modification. Only one table can be selected at a time.
B (BROWSE)	Display a list of jobs in a table for browsing. Only one table can be browsed at a time.
O (ORDER)	Order all the jobs in the table, provided that their basic scheduling criteria, as described in "Basic Scheduling Parameters" on page 131, are satisfied. Multiple tables can be ordered.
	When ordering multiple tables from the Table List Screen with one confirmation window for all tables (that is, ASK FOR EACH ONE =N), the user must exit from Screen 2.0 for each table to ensure that all the requested tables are ordered.

Table 33 Options of the Table List Screen (part 2 of 2)

Option	Description
F (FORCE)	Order all the jobs in the table, regardless of their basic scheduling parameters, which are described in "Basic Scheduling Parameters" on page 131. Multiple tables can be forced.
G (GRAPHIC) FLOW	Display a graphic presentation of the job flow of the jobs in the table, as described in "Displaying Graphic Jobflow" on page 162. Only one table at a time can be selected for graphic display.
D (DELETE)	Delete the table (member) from the library. Multiple tables can be deleted, as described in "Deleting Tables" on page 158.

### - NOTE



If your access to options has been limited by the INCONTROL administrator, you can only access the BROWSE option.

## **Commands of the Table List Screen**

The following command can be entered in the **COMMAND** field of the Table List screen.

### The SELECT Command

The SELECT command can be used to create a new table in the library. The format of the command is:

SELECT tablename type

### Valid types are:

- GRP For Group scheduling tables.
- JOB For regular scheduling tables.

If no type is entered, the default type is JOB.

#### – NOTE –



If the SELECT command is entered for an existing table, it acts like the S (SELECT) line option, which is described in Table 33, and displays the list of jobs in the table.

If there are no jobs currently in the table, that is, the command is being used to create a new table, the Job List screen is not displayed. Instead

- A skeletal job scheduling definition is displayed in the Job Scheduling Definition screen if a Job scheduling table is being created.
- A skeletal Group Entity scheduling definition is displayed in the Scheduling Definition screen if a Group scheduling table is being created.

## **Job List Screen**

The Job List screen displays the list of jobs in a scheduling table in a specified library. This screen can be entered directly from the entry panel or the Table List screen, or upon exiting from the Job Scheduling Definition screen.

#### NOTE



The names displayed on the Job List screen are the names of the members that contain the JCL of the jobs, which is specified in the MEMNAME parameter in the job scheduling definition, or, in the case of started tasks, the name of the STC.

If the S (Select) option was entered in the Table List screen for a table that is currently in use ("selected") by another user, either the Job List screen is not displayed and the Table List screen remains displayed (the default), or the Job List screen is displayed in Browse mode (if a user profile definition overrides the default). In either case, an appropriate message is displayed.

Figure 16 CONTROL-M Scheduling Definition Facility Job List Screen

```
JOB LIST
            LIB: CTM.PROD.SCHEDULE
                                                              TABLE: BACKUP
COMMAND ===>
                                                              SCROLL===> CRSR
OPT NAME ---
              TYP --- DESCRIPTION ---- GROUP: GRPWK1 --
     STARTBKP
               G START OF DAILY BACKUP
              J DAILY BACKUP OF DATA SETS FROM APPL-L
     BACKPL01
     BACKPL02
                  DAILY BACKUP OF SPECIAL FILES FROM APPL-L
               J WEEKLY BACKUP OF FILES FROM APPL-L #1
     BACKPLW1
               J WEEKLY BACKUP OF FILES FROM APPL-L #2
     BACKPLW2
              J WEEKLY BACKUP OF FILES FROM APPL-L #3
     BACKPLW3
     BACKPLW4 J WEEKLY BACKUP OF FILES AFTER DAILY FROM APPL-L +
     DASDRPT1 J DASD REPORTS AFTER BACKUPS FOR APPL-L
     DASDRPT2 J DASD STATISTICS REPORT AFTER BACKUP FOR APPL-L
     ENDPLBKP J END OF BACKUP INDICATION FOR APPL-L
     BACKACO1 J DAILY BACKUP OF DATA SETS FROM APPL-ACCOUNT
     BACKACO2 J DAILY BACKUP OF SPECIAL FILES FROM APPL-ACCOUNT
     BACKACW1 J WEEKLY BACKUP OF FILES FROM APPL-ACCOUNT #1
     BACKACW2 J WEEKLY BACKUP OF FILES FROM APPL-ACCOUNT #2
     BACKACW3 J WEEKLY BACKUP OF FILES FROM APPL-ACCOUNT #3
     BACKACW4 J WEEKLY BACKUP OF FILES AFTER DAILY FROM APPL-ACC +
     DASDRPT3 J DASD REPORTS AFTER BACKUPS FOR APPL-ACCOUNT
     DASDRPT4
                  DASD REPORTS AFTER BACKUP FOR APPL-ACCOUNT
                  END OF BACKUP INDICATION FOR APPL-ACCOUNT
     ENDACBKP
     BACKDD01
              J
                  DAILY BACKUP OF DATA SETS FROM APPL-DD
OPTIONS S SEL D DEL I INS O ORDER F FORCE J JCL C COPY P PLN T JOBSTAT 15.37.39
```

## Format of the Job List Screen

Next to each job name in the Job list, certain information can be displayed. The type and format of this information depends on whether the screen is displayed in DESC format, in DATA format or in STAT format, and whether the list is displayed for a Group scheduling table, as follows:

- In DESC format, the description of the job, taken from the **DESC** field of the job scheduling definition, is displayed. Default.
- In DATA format, the application and group names of the job, taken from the APPL and GROUP fields of the job scheduling definition, are displayed.
- In STAT format, ISPF-like statistical information about the job is displayed.
- If the job list is displayed for a Group scheduling table, the type of job scheduling definition is also displayed in the DESC, DATA, and STAT formats. Type information is not displayed for regular scheduling tables. Valid values are:
  - G Group Entity; this is always the first entry in the Job list
  - J Job

By default, the job list is displayed in DESC format. To change formats, use the DESC, DATA or STAT commands, described below.

The order in which the jobs are displayed in the Job List screen can be sorted by the SORT command (described below).

## **Commands of the Job List Screen**

The following commands can be entered in the **COMMAND** field of the Job List screen:

Table 34 Commands of the Job List Screen (part 1 of 2)

Command	Description
DESC	Command DESC displays the job description next to the job name. The description is taken from the <b>DESC</b> field in the job scheduling definition.
DATA	Command DATA displays the Application name and Group name of the job next to the job name. The Application name and Group name are taken from the corresponding fields in the job scheduling definition.
STAT	Command STAT displays (next to the job name) the following ISPF-like statistical information about the job: version and modification numbers, creation date, last modification date, and user ID.

Table 34 Commands of the Job List Screen (part 2 of 2)

Command	Description
SORT	Command SORT sorts the list of jobs in the Job List screen according to specified criteria. Format of the command is:
	SORT key
	Where key is one of the following values:
	<ul> <li>J (Job) – Sorted according to job name.</li> <li>G (Group) – Sorted according to group name.</li> <li>A (Application) – Sorted according to application name.</li> </ul>

## **Options of the Job List Screen**

To request one of the following options, type the option in the **OPT** field to the left of the job name and press **Enter**.

#### NOTE -



Option O (Order) is not available if the Job List screen is displayed for a Group scheduling table.

Options I (Insert) and J (JCL) are not available for Group Entities.

If the Job List screen is displayed in Browse mode, options D (Delete) and I (Insert) are not available.

Table 35 Options of the Job List Screen (part 1 of 2)

Option	Description
S (SEL)	Display the Job Scheduling Definition screen, with details of the selected job. Only one job can be selected at a time.
	If the Job List screen is not displayed in Browse mode, the job scheduling definition can be edited and updated. If the Job List screen is displayed in Browse mode, the job scheduling definition can only be browsed; it cannot be modified.
D (DEL)	Delete a job from the Job list (member). Multiple jobs can be selected.  See "Condition Inheritance" on page 159 for a description of an optional feature in CONTROL-M that automatically adjusts all the jobs in the table so that the jobs that are waiting for the deleted job's OUT conditions will inherit its IN conditions.

Table 35 Options of the Job List Screen (part 2 of 2)

Option	Description
I (INS)	Insert a new job in the list (member). The Job Scheduling Definition screen appears, with the same details of the job marked "I", but the MEMNAME and DESCRIPTION parameters are empty for you to fill in. The new job is added after the job marked "I". Only one new job can be inserted at a time.
O (ORDER)	Order a job provided that its basic scheduling criteria, as described in "Basic Scheduling Parameters" on page 131, are satisfied. Multiple jobs can be selected.
F (FORCE)	Force a job order regardless of the basic scheduling criteria of the job, as described in "Basic Scheduling Parameters" on page 131. Multiple jobs can be selected.
J (JCL)	Edit the member that contains the JCL of the job. Entering this option brings you directly into the JCL member in ISPF Edit mode. By default, if the JCL member exists in the OVERLIB library, that member is edited. If the JCL member does not exist in the OVERLIB library, the member is edited in the MEMLIB library. You can only edit the JCL of one job.
	For more information on the OVERLIB and MEMLIB libraries, see "OVERLIB: General Job Parameter" on page 571 and "MEMLIB: General Job Parameter" on page 523.
	In ISPF Edit mode, if the name in the MEMNAME parameter contains a mask character (for example, if it is a generic name such as PRODJOB*), using the J option displays all PDS members in the library with names that match the mask.
C (COPY)	Copy the job to another table. Multiple jobs can be selected. For more information, see "Copying Jobs to Another Table" on page 156.
P (PLN)	Display a schedule plan for the job. You can only display the schedule plan of one job. For more information, see "Displaying a Job Scheduling Plan" on page 164.
T (JOBSTAT)	Displays the statistics for the job in the CONTROL-M Statistics screen, described in "Statistics Screen" on page 238.
% (AUTOEDIT SIMULATION)	Simulates the action of the CONTROL-M Submission mechanism for the member and JCL library specified in the MEMNAME and MEMLIB parameters of the job scheduling definition. This option operates in the same way as the % option in the Active Environment screen (see Table 58 on page 180 for full details).
	<b>Note:</b> When requesting the '%' option, a prompt for the simulation ODATE is displayed.

## Job Scheduling Definition Screen - Defining Schedules

The CONTROL-M Job Scheduling Definition screen is used to define, display and modify production parameters of a specific job. This screen can be entered directly from the entry panel or from the Job List screen. Update of parameters is not permitted in Browse mode.

### NOTE -



The format of the Job Scheduling Definition screen for Group Entities is slightly different than the format shown below and is described in "Scheduling Definition for Group Entities" on page 138.

### Figure 17 Job Scheduling Definition Screen

```
JOB: BACKPLO2 LIB CTM.PROD.SCHEDULE
                                                                       TABLE: BACKUP
                                                                       SCROLL===> CRSR
COMMAND ===>
 MEMNAME BACKPLO2 MEMLIB CTM.PROD.JOBLIB
OWNER M44 TASKTYPE JOB PREVENT-NCT2 Y DFLT N
APPL APPL-L GROUP BKP-PROD-L
  DESC DAILY BACKUP OF SPECIAL FILES FROM APPL-L
  OVERLIB CTM.OVER.JOBLIB
                                                                 STAT CAL
                                   SYSTEM ID
                                                                  NJE NODE
  SCHENV
  SET VAR
  CTB STEP AT NAME TYPE
DOCMEM BACKPLO2 DOCLIB CTM.PROD.DOC
  SCHEDULE TAG
  RELATIONSHIP (AND/OR) O
  DAYS ALL
                                                                       DCAL
                                                                             AND/OR
                                                                       WCAL
  MONTHS 1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- Y 11- Y 12- Y
  DATES
                     SHIFT RETRO N MAXWAIT OO D-CAT
```

MINIMUM DEFINITION ACT	PDS IVE FROM	UNTIL			
IN START CONTROL	- DAILY-BACKUP	ODAT			
RESOURCE INIT		0001	CART	000	1
	ROD.PIPE				
FROM TIME		UNTIL TIN	· <del>-</del>		
DUE OUT TIME TIME ZONE:	+ DAYS	PRIURITY	SAC	CUNFIRM	
======================================	 :PL02-ENDED-OK	ODAT +			
AUTO-ARCHIVE Y					
	OF DAYS TO KEEP	# 0F (	GENERATIONS	TO KEEP	
	(,D,F,N,R)		T.A.	ITED.VAI	FROM
MAXRERUN STEP RANGE	FR (PGM.PF	200)	1 N	ITERVAL TO	FROM
ON PGMST	PROCST	CODES	•	10	A/O
DO DO	1110001	00020			71,70
ON SYSOUT				FROM 001 TO 1	32 A/0
DO DO					
SHOUT WHEN	TIME	+ DAY	'S TO		URGN
MS 					
APPL TYPE			APPL VER	?	
APPL FORM			CM VER	R	
INLINE JCL: N					
	>>>>>> EN[		ING PARAMET	ERS <<<<<<	
UMMANDS: EDIT,	DOC, PLAN, JOE	SSIAI			11.27.4

#### - NOTE



The SCHEDULE TAG and RELATIONSHIP parameters appear only in job scheduling definitions belonging to Group scheduling tables.

The PIPE parameter is displayed only if MAINVIEW Batch Optimizer (MVBO) is installed.

RETENTION parameters # OF DAYS TO KEEP and # OF GENERATIONS TO KEEP are displayed only at sites that use the History Jobs file.

The job scheduling definition occupies more than one screen.

To delete a parameter on the screen, simply erase it with the EOF key or blank it out. If additional action is required, CONTROL-M issues appropriate instructions.

## Parameters of the Job Scheduling Definition Screen

The Job Scheduling Definition screen is divided into the following sections.

Table 36 Parameters of the Job Scheduling Definition Screen

Parameter	Description
General Job Parameters	In this section, you can enter specific information about the job itself – in which member and library the JCL is found, who is the owner of the job, and so on.
Basic Scheduling Parameters	In this section, you can enter scheduling criteria (for example, the days of the week or month on which the job must be submitted).
Runtime Scheduling Parameters	In this section, you can enter submission criteria including conditions that must be fulfilled (generally, successful completion of a preceding job) before submission of the job, resources required by the job, and time limitations on job submission.
Post-Processing Parameters	In this section, you can enter fixed or conditional actions to perform upon job completion, or upon the execution of specified job steps. For example, you can set conditions that trigger the submission of other jobs, you can send messages to the operator console, or you can rerun the job.

These sections are divided by a delimiter line.

A brief description of all parameters in each section of the Job Scheduling Definition screen is provided on the following pages. A detailed explanation of these parameters is provided in Chapter 3, "Job Production Parameters."

#### NOTE -



Parameters marked with the symbol M can have multiple occurrences. Whenever you fill in the last occurrence of the parameter on the screen, CONTROL-M adds a new empty occurrence of the parameter that you may fill in. The only limit to the number of occurrences is the region size available for the application.

## **General Job Parameters**

Figure 18 General Job Parameters

```
MEMNAME BACKPLO2 MEMLIB CTM.PROD.JOBLIB
OWNER M44
                   TASKTYPE JOB PREVENT-NCT2 Y DFLT N
APPL APPL-L
                                   GROUP BKP-PROD-L
DESC
     DAILY BACKUP OF SPECIAL FILES FROM APPL-L
OVERLIB CTM.OVER.JOBLIB
                                                         STAT CAL
SCHENV
                              SYSTEM ID
                                                         NJE NODE
SET VAR
CTB STEP AT
                    NAME
DOCMEM BACKPL02
                    DOCLIB CTM.PROD.DOC
```

Table 37 General Job Parameters (part 1 of 2)

Parameter	Description	
MEMNAME	Name of the member that contains the JCL of the job, or name of the started task.	
MEMLIB	Name of the library that contains either the JCL of the job or identifying information and parameters of the started task.	
OWNER	ID of a user who requests CONTROL-M services. This field is used for security purposes.	
TASKTYPE	Type of task to be performed by CONTROL-M (for example, job – JOB, started task – STC).	
§Restart§ PREVENT-NCT2	Indicator (Y/N/F/L) specifying whether (and how) to perform data set cleanup prior to the original job submission.	
	The subparameter <b>DFLT</b> is a protected field that indicates the PREVENT-NCT2 default value at this site.	
APPL	Name of the application to which the group of the job belongs.	
GROUP	Name of the group to which the job belongs, such as BACKUPS, RESERVATIONS, INVENTORY, and so on.	
DESC	Description of the job (free text) that is displayed next to the job name in the Job List screen.	
OVERLIB	Name of a library that overrides the library specified in MEMLIB.	
STAT CAL	Name of a periodic calendar that will be used to gather average runtime statistics for the job, based on a period.	
SCHENV	Name of the workload management scheduling environment to be associated with the job.	
SYSTEM ID	In JES2, the identity of the system in which the job must be initiated and executed.	
	In JES3, the identity of the processor on which the job must be executed.	
NJE NODE	Identifies the node in the JES system at which the job must execute.	
SET VAR ^M	Statement assigning a value to an AutoEdit variable, which can be used in the submitted job.	
CTB STEP	CONTROL-M/Analyzer definition to be activated as the first or last (as entered) step of the job. The type of CONTROL-M/Analyzer definition (rule or mission) and its name are also entered.	
DOCMEM	Name of a member in which the job documentation resides.	
DOCLIB	Name of the library in which the job documentation member resides.	
DOC	Detailed job documentation.	
INSTREAM JCL	Whether CONTROL-M for $z/OS$ submits a JCL stream defined within the job scheduling definition.	

Table 37 General Job Parameters (part 2 of 2)

Parameter	Description		
The following General	The following General parameters are in the Group Entity only:		
ADJUST CONDITIONS	Allows conditions to be removed from job orders if the predecessor jobs that set the conditions are not scheduled.		
GRP MAXWAIT	Number of additional days after the original scheduling date that the Group Entity can remain in the Active Jobs file if it does not have a status of ENDED OK (and none of its jobs currently appear in the Active Jobs file).		

## **Basic Scheduling Parameters**

Figure 19 Basic Scheduling Parameters - Job

```
SCHEDULE TAG
RELATIONSHIP (AND/OR) O
                                                            DCAL
DAYS
                                                                  AND/OR
                                                            WCAL
WDAYS
MONTHS 1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- Y 11- Y 12- Y
DATES
                            RETRO N MAXWAIT 04 D-CAT
CONFCAL
                 SHIFT
MINIMUM
                 PDS
DEFINITION ACTIVE FROM
                               UNTIL
```

Figure 20 Basic Scheduling Parameters - Group Entity

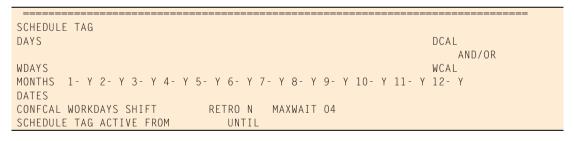


Table 38 Basic Scheduling Parameters (part 1 of 3)

Parameter	Description		
The following parameters apply only when defining a Group Entity. For more information on how these parameters are applied to jobs in Group Tables, see Chapter 3, "Job Production Parameters."			
SCHEDULE TAG ^M Tags identifying the sets of scheduling criteria defined in the Entity that can be used to schedule the job.			
RELATIONSHIP	And/Or indicator that determines whether or not the criteria of the specified schedule tag and the basic scheduling criteria of the individual job must both be satisfied.		

 Table 38
 Basic Scheduling Parameters (part 2 of 3)

Parameter	Description	
DAYS	Days of the month to schedule the job. A maximum of two lines can be entered. Subparameters are:	
	<ul> <li>DCAL – identifies a DAYS calendar containing predefined scheduling dates</li> </ul>	
	■ AND/OR – conjunctional subparameter used to link the DAYS and WDAYS parameters when both are scheduled	
WDAYS	Days of the week to schedule the job. A maximum of two lines can be entered.	
	The WCAL subparameter identifies a calendar containing working days on which a job can be scheduled.	
MONTHS	Months to run the job.	
DATES	Specific dates in the year to run the job.	
CONFCAL	Name of a calendar used to confirm job scheduling dates.  The optional subparameter SHIFT indicates when and if a job must	
RETRO	be scheduled.  Yes or No (Y/N) indicator specifying whether the job is to be scheduled (retroactively) after the original scheduling date has passed.	
MAXWAIT	Number of extra days within which to try to execute a job in the Active Jobs file if the date of the job has passed.	
D-CAT	Name of a CONTROL-D report decollating mission category. If specified, the report decollating mission is scheduled whenever the job is scheduled under CONTROL-M.	
MINIMUM	Minimum number of free tracks required by the library specified in the PDS parameter. The job is executed if the number of free tracks is less than the minimum.	
PDS	Name of a partitioned data set to be checked for free space. If the PDS has less than the minimum number of required free tracks, as specified in the MINIMUM parameter, the job is executed. Not supported for PDSE-type libraries.	

 Table 38
 Basic Scheduling Parameters (part 3 of 3)

Parameter	Description				
	eters apply only when defining individual jobs. For more information see Chapter 3, "Job Production Parameters."				
DEFINITION ACTIVE FROM	When a date is entered in this field within a job scheduling definition, the job will only be ordered if the current date is later th that date. Valid values are:				
	<ul> <li>Date in the format ddmmyy, or mmddyy, or yymmdd, depending on your site format, as set by the DATETYP parameter in the IOAPARM member in the IOA PARM library.</li> </ul>				
	■ ''(Blank)				
DEFINITION ACTIVE UNTIL	When a date is entered in this field within a job scheduling definition, the job will only be ordered if the current date is earlier than that date. Valid values are:				
	<ul> <li>Date in the format ddmmyy, or mmddyy, or yymmdd, depending on your site format, as set by the DATETYP parameter in the IOAPARM member in the IOA PARM library.</li> </ul>				
	• ' ' (Blank)				
	eters apply only when defining a Group Entity. For more information ters are applied to jobs in Group Tables, see Chapter 3, "Job Production"				
SCHEDULE TAG ACTIVE FROM	When a date is entered in this field within a Group scheduling definition, a job which refers to this Schedule Tag will only be ordered if the current date is later than that date. Valid values are:				
	■ Date in the format ddmmyy, or mmddyy, or yymmdd, depending on your site format, as set by the DATETYP parameter in the IOAPARM member in the IOA PARM library.				
	• ''(Blank)				
SCHEDULE TAG ACTIVE UNTIL	When a date is entered in this field within a Group scheduling definition, a job which refers to this Schedule Tag will only be ordered if the current date is earlier than that date. Valid values are:				
	■ Date in the format ddmmyy, or mmddyy, or yymmdd, depending on your site format, as set by the DATETYP parameter in the IOAPARM member in the IOA PARM library.				
	■ ''(Blank)				

## **Runtime Scheduling Parameters**

Figure 21 Runtime Scheduling Parameters

```
IN START-DAILY-BACKUP ODAT

CONTROL

RESOURCE INIT 0001 CART

PIPE

FROM TIME + DAYS UNTIL TIME + DAYS

DUE OUT TIME + DAYS PRIORITY SAC CONFIRM

TIME ZONE:
```

**Table 39 Runtime Scheduling Parameters** 

Parameter	Description	
$IN^M$	Prerequisite conditions for the job.	
CONTROL ^M	Shared or exclusive control over resources required for the job.	
RESOURCE ^M	Quantitative resources required for the job.	
PIPE	Name of a data set replaced by a pipe during the run of the job. Available only at sites in which MAINVIEW Batch Optimizer (MVBO) is installed.	
TIME + DAYS	Time limit (FROM, UNTIL) for job submission.	
PRTY	Job priority in receiving CONTROL-M services or critical path priority.	
CONFIRM	Yes or No indicator $(Y/N)$ specifying whether manual confirmation is required before the job can be submitted.	
DUE OUT + DAYS	Time by which the job must finish executing.	
SAC	Enables scheduled runs of a job that was converted from another job scheduling product, such as CA-7, to be shifted to their proper scheduling days.	
TIME ZONE	Enables automatic adjustment of the times specified in the job definition to the corresponding times in a different time zone.	

## **Post-Processing Parameters**

Figure 22 Post-Processing Parameters

```
OUT BAKCKPL02-ENDED-OK ODAT +

AUTO-ARCHIVE Y SYSDB Y MAXDAYS MAXRUNS

RETENTION: # OF DAYS TO KEEP # OF GENERATIONS TO KEEP

SYSOUT OP (C,D,F,N,R) FROM

MAXRERUN RERUNMEM INTERVAL FROM

STEP RANGE FR (PGM.PROC) . TO .

ON PGMST ANYSTEP PROCST CODES CO008 U0048 A/O

DO
```

Table 40 Post-Processing Parameters (part 1 of 2)

Parameter	Description
$\mathrm{OUT^M}$	Prerequisite conditions to be added and/or deleted when the job finishes OK.
§Restart§ AUTO-ARCHIVE	Yes or No indicator (Y/N) specifying whether to automatically archive SYSDATA. Subparameters are:
	<ul> <li>SYSDB – Yes or No indicator specifying whether to archive SYSDATA of jobs to a common data set (Y) or to unique data sets (N)</li> </ul>
	■ MAXDAYS – maximum number of days (from 00 through 98, or 99) to retain archived SYSDATA of jobs that ended NOTOK
	■ MAXRUNS – maximum number of runs (from 000 through 999) for which the archived SYSDATA must be retained for jobs that ended NOTOK
RETENTION	Either of the following subparameters (but not both) can be used to specify how long the job must remain in the History Jobs file:
	■ # OF DAYS TO KEEP – number of days the job must be retained
	<ul> <li># OF GENERATIONS TO KEEP – number of runs of the job that must be retained</li> </ul>
SYSOUT	Action to perform with the job sysout when the job finishes OK.
MAXRERUN	Maximum number of reruns to be performed for the job, if DO RERUN is requested. (Called RERUN – MAXRERUN prior to version 6.0.00.)
RERUNMEM	Name of member to be submitted in case of a DO RERUN request. (Called RERUN – RERUNMEM prior to version 6.0.00.)
INTERVAL	Amount of time, in minutes, to wait between reruns or between cycles of a cyclic job.
STEP RANGE ^M	Step range (FROM or TO PGMST, and optionally PROCST) name to be referenced by ON statements.
$ON^M$	Step and code event criteria that determine if the accompanying DO actions are performed
	■ PGMST/PROCST – program step (and optionally the procedure step) to check for the specified code criteria
	■ CODES ^M – execution event codes, such as U0234, SB37, and C0004
	■ A/O – AND/OR conjunctional parameter that opens (and links) additional ON statements

Table 40 Post-Processing Parameters (part 2 of 2)

Parameter	Description	
DO action ^M	Actions to be performed when the ON step/code event criteria are satisfied. Valid DO actions are described in "DO Actions" below.	
SHOUT ^M	<ul> <li>Message to be sent to a specified destination in a specified situation:</li> <li>WHEN – Situations under which to send the message.</li> <li>TO – Destination of the message.</li> <li>URGN – Urgency of message.</li> <li>MS – The message, in free text. CONTROL-M AutoEdit System variables are supported.</li> </ul>	

### **DO Actions**

The following are a description and example of each of the DO actions.

**DO COND** – Add or delete a prerequisite condition.

ON PGMST UPDATE	PROCST	CODES S*** U****	A/0
DO COND	PL-BAC	KOUT-REQUIRED ODAT +	
DO DO			

**DO CTBRULE** – Invoke the CONTROL-M/Analyzer Runtime environment and execute the specified CONTROL-M/Analyzer rule, which performs the balancing operations defined in the rule on SYSDATA. Arguments can be passed to CONTROL-M/Analyzer. Available only at sites in which CONTROL-M/Analyzer is active.

ON PGMST ANYSTEP	PROCST	CODES OK	A/0
DO CTBRULE	= BALKPL	ARG DOREPORT, UPDATEDB	
DO			

### **DO FORCEJOB** – Force (schedule) a job under CONTROL-M.

ON PGMST UPDATE PROCST	CODES S*** U****	A/0
DO FORCEJOB TABLE PLPROD	JOB PLBCKOUT	DATE ODAT
LIBRARY GENERAL		
DO		

**§Restart§ DO IFRERUN** – Perform a restart under CONTROL-M/Restart when the job is manually or automatically rerun. This DO action is available if you have CONTROL-M/Restart installed at your site.

ON PGMST ANYSTEP	PROCST	CODES SOC1	A/0
DO IFRERUN	FROM GLSTEP01	. GLPROCO2 TO GLSTEP05	. GLPROCO3 CONFIRM N
DO			

## DO MAIL - Send an e-mail message to the specified recipients.

ON PGMST UPDATE	PROCST	CODES COOOO	A/0
DO MAIL			
TO ACCT-SMITH			
CC			
SUBJ VERIFICATION			
TEXT CHECK OK			
D0			

## **DO NOTOK** – Define the termination status of the job as NOTOK.

ON PGMS	UPDATE	PROCST	CODES COOO4	A/0
1 OD	NOTOK			
DO				

## **DO OK** – Define the event within the job as OK.

ON PGMST ANYSTEP	PROCST	CODES COOO8	U0048	A/0
DO OK				
DO				

## **DO REMEDY** – Open a Remedy Help Desk ticket.

0	N PGMST STEP1	PROCST	CODES COO12	A/0
D	OO REMEDY			URGENCY U
S	SUMM JOB %%JOBNAME	ON NODE %%\$NODEID	ENDED WITH RETURN CO	ODE %%COMPSTAT
D			I ENDED ON NODE %%\$N0 GAPPL, GROUP: %%GROUI	

### — NOTE –



To close the ticket, the CONTROL-M user will have to access the Remedy online services.

## **DO RERUN** – Indicate that a job must be rescheduled for a rerun.

ON PGMST ANYSTEP PROCST	CODES SOC1	A/0
DO IFRERUN FROM GLSTEP01	. GLPROCO2 TO GLSTEPO5 . GLPR	OCO3 CONFIRM N
DO RERUN		
DO		

## **DO SET** – Assign a value to a CONTROL-M AutoEdit variable.

ON PGMST ANYSTEP PROCST	CODES COOO8	U0048	A/0
DO SET VAR= %%RUNTYPE=CHK			
DO			

### – NOTE



Since DO SET values are dependent upon fulfillment of ON step or codes criteria, the values are assigned after job execution and used for subsequent cyclic runs and rerun.

### **DO SHOUT** – Message to be sent to specified location.

Ī	ON PGMST UPDATE	PROCST	CODES S*** U****	A/0
	DO SHOUT TO	OPER2	URGENCY R	
	= A BACKOUT OF	FILE-XXXX IS	GOING TO RUN SOON	
	DO SHOUT TO	USER-DBA	URGENCY R	
	= ABEND OF THE	UPDATE STEP,	PLEASE CHECK IT	
	DO DO			

### **DO STOPCYCL** – Stop recycling a cyclic task.

ON PGMST ANYSTEP	PROCST	CODES >COOO8	A/0
DO STOPCYCL			
DO			

## **DO SYSOUT** – Action to perform with the job sysout.

ON PGMST UPDATE PROCST	CODES S*** U****	A/0
DO SYSOUT OPT C PRM P		FRM
DO		

## **Scheduling Definition for Group Entities**

A Group Entity must be defined for each Group scheduling table before the job scheduling definitions in the table can be defined.

A skeletal scheduling definition for a Group Entity is automatically displayed when creating a Group scheduling table.

The scheduling definition for a Group Entity can also be entered directly from the Entry Panel or from the Job List screen.

The job scheduling definition for Group Entities varies somewhat from the job scheduling definition for jobs.

The parameters of the Group Entity are used to define basic scheduling criteria, runtime scheduling criteria, and post-processing actions to be performed, for the jobs in the group.

During New Day processing, if at least one set of basic scheduling criteria in the Group Entity is satisfied, a copy of the Group Entity is placed in the Active Jobs file, and the jobs in the Group Entity become eligible for scheduling.

The final status of the Group Entity job order is assigned after all scheduled jobs in the table have been terminated. This Group Entity status is determined by the execution results of those jobs:

- If all the scheduled jobs in the table ended OK, the Group Entity is assigned an end status of OK.
- If at least one scheduled job in the table did not end OK, the Group Entity is assigned an end status of NOTOK.

The performance of post-processing actions defined in the Group Entity is directly affected by the end status of the Group Entity.

Figure 23 Group Entity Scheduling Definition Screen

```
GRP ACCOUNTS GROUP CTM.PROD.SCHEDULE(GRP)
COMMAND ===>
                                                  SCROLL===> CRSR
+-----
 GROUP ACCOUNTS_GROUP MEMNAME ACCOUNTS
 OWNER NO4B
 APPL
 DESC
 ADJUST CONDITIONS N
                         GRP MAXWAIT
                                             STAT CAL
  SET VAR
 DOCMEM ACCOUNTS DOCLIB CTM.PROD.DOC
        _____
 SCHEDULE TAG ALL_DAYS
 DAYS ALL
                                                  DCAL
                                                      AND/OR
                                                  WCAL
 MONTHS 1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- Y 11- Y 12- Y
         SHIFT
                      RETRO N MAXWAIT 00
 SCHEDULE TAG ACTIVE FROM UNTIL
 SCHEDULE TAG
 DAYS
                                                  DCAL
                                                      AND/OR
 WDAYS
 MONTHS 1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- Y 11- Y 12- Y
         SHIFT
                       RETRO N MAXWAIT 00
 CONFCAL
 SCHEDULE TAG ACTIVE FROM UNTIL
 CONTROL
 RESOURCE
 PIPE
 FROM TIME + DAYS UNTIL TIME + DAYS
DUE OUT TIME + DAYS PRIORITY SAC CONFIRM
 TIME ZONE:
 OUT
 ON GROUP-END NOTOK
   DO COND ACCTS-CHK-REQUIRED ODAT +
HOUT WHEN TIME + DAYS
 SHOUT WHEN
                 TIME + DAYS TO
                                                       URGN
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
                                                         18.19.14
```

The Group Entity scheduling definition supports the same commands and PFKey conventions as any job scheduling definition.

# Parameters of the Group Entity Scheduling Definition Screen

The parameters of the Group Entity scheduling definition are almost identical in appearance and usage as those in a regular job scheduling definition, which are described briefly in "Parameters of the Job Scheduling Definition Screen" on page 128.

Differences in the parameters of the Group Entity scheduling definitions are described below.

All parameters of the Job Scheduling Definition screen are described in detail in Chapter 3, "Job Production Parameters."

Table 41 Parameters of the Group Entity Scheduling Definition Screen (part 1 of 2)

Parameter	Description
GROUP	Name of the group. This parameter also appears, in a different location, in regular job scheduling definitions. Mandatory in Group Entities. When defined in a Group Entity, the value is automatically assigned to all job scheduling definitions in the Group table.
MEMNAME	In a Group Entity, this field does not indicate a member name.  Instead, it is used to indicate an abbreviated group name. This abbreviated group name is then displayed, when appropriate, in other screens, such as the Active Environment screen.
ADJUST CONDITIONS	Yes or No indicator specifying whether prerequisite conditions normally set by predecessor jobs are removed from job orders if the relevant predecessor jobs are not scheduled. This parameter only appears in the Group Entity, and applies to all jobs in the table awaiting prerequisite conditions from unscheduled jobs. Use of this parameter can simplify the handling of Maybe jobs.
GRP MAXWAIT	Number of extra days beyond the original scheduling date the Group Entity can be maintained in the Active Jobs file if it does not have a status of ENDED OK.  However, if one of its jobs remains in the Active Jobs file beyond this number of days, the Group Entity remains in the Active Jobs file as long as the job remains there.

Table 41 Parameters of the Group Entity Scheduling Definition Screen (part 2 of 2)

Parameter	Description
SCHEDULE TAG	Unique identifier to be applied to the accompanying set of scheduling criteria. Multiple sets of scheduling criteria can be defined, each with its own tag.
	A set of criteria defined in a Group Entity can be applied to a job by specifying the identifying tag in the job scheduling definition.
	At least one set of basic scheduling criteria in the Group Entity must be satisfied before the jobs in that Group scheduling table become eligible for scheduling on any day.
Basic Scheduling Parameters	The Group Entity does not contain parameters D-CAT, PDS, and MINIMUM, which are found in job scheduling definitions.
Runtime Scheduling Parameters	These parameters, IN, TIME FROM/UNTIL, PRIORITY, DUE OUT, and CONFIRM, apply to all scheduled jobs in the group.
	All runtime scheduling criteria in the Group Entity must be satisfied before any of the scheduled jobs are eligible for submission. Any runtime criteria defined for a particular job must also be satisfied before the job can be submitted.
Post-Processing Parameters	Non-conditional Post-processing parameters (OUT, SHOUT) are performed only if all scheduled jobs in the table have ended OK.
	ON GROUP-END Group Entity end status indicator. This parameter appears only in the Group Entity. DO statements immediately following this parameter are performed only if the Group Entity is assigned the indicated status. Valid values are:
	<ul> <li>OK – Subsequent DO actions are performed for each job in the group only if the end status of the Group Entity is OK (that is, all scheduled jobs in the table ended OK).</li> </ul>
	■ NOTOK – Subsequent DO actions are performed for each job in the group if the end status of the Group Entity is NOTOK (that is, at least one job in the group ended NOTOK).

## **Commands of the Job Scheduling Definition Screen**

The following commands can be entered in the **COMMAND** field of the Job Scheduling Definition screen.

Table 42 Commands of the Job Scheduling Definition Screen (part 1 of 2)

Command	Description	
EDIT	Alternately places the job scheduling definition in, and removes the job scheduling definition from, an ISPF-like Edit environment.	
	For a brief overview, see "Editing Job Scheduling Definitions in the Edit Environment" on page 143. For more complete details, see Appendix A, "The CONTROL-M Application Program Interface (CTMAPI)."	
DOC	Alternately displays and hides the job documentation.	
	For more information, see "Job Documentation" on page 144.	
PLAN	Enables display of the job's scheduling plan.	
	When the PLAN command is entered, a window for entering the date range of the plan is displayed. When the date range is entered, the scheduling plan for the job is displayed in the Job Scheduling Plan screen.	
	For more information, see "Displaying a Job Scheduling Plan" on page 164.	
JOBSTAT	Displays the Statistics screen, which provides statistics for the job.	
	To display statistics for the currently displayed job, type:	
	JOBSTAT (abbreviated J)	
	To display statistics for any job other than the current job, format of the command is:	
	JOBSTAT jobname groupname	
	Entering a group name is optional, but if no group name is entered, statistics are displayed only for jobs not belonging to any group.	
	For more information, see "Statistics Screen" on page 238.	

Table 42 Commands of the Job Scheduling Definition Screen (part 2 of 2)

Command	Description
CHANGE	Replaces an existing string with a new string.
	The format of the command is:
	CHANGE oldstring newstring
	where:
	<ul> <li>oldstring is the existing string to be replaced</li> <li>newstring is the string that replaces the existing string</li> </ul>
NEXT	The NEXT command ( <b>PF11/PF23</b> ) keeps the changes to the current job scheduling definition in memory and automatically displays the next job scheduling definition in the scheduling table.
	For more information, see "Exiting the Job Scheduling Definition Screen" on page 148.
PREV	The PREV command ( <b>PF10/PF22</b> ) keeps the changes to the current job scheduling definition in memory and automatically displays the previous job scheduling definition in the scheduling table.
	For more information, see "Exiting the Job Scheduling Definition Screen" on page 148.

If a string contains embedded spaces, enclose the string in apostrophes (') or quotation marks (").

To repeat a CHANGE command, press PF09/PF21.

## **Editing Job Scheduling Definitions in the Edit Environment**

Job scheduling definition parameters can be edited, that is, moved, copied, deleted, or repeated, by performing IOA Line Editing commands, similar to standard ISPF line commands, from within the CONTROL-M Edit environment.

The Edit Environment in the Job Scheduling Definition screen is accessed by typing **EDIT** in the **COMMAND** field and pressing **Enter**.

CONTROL

COMMANDS: EDIT, DOC, PLAN, JOBSTAT

JOB: BACKPO2 LIB CTM.PROD.SCHEDULE TABLE: BACKUP COMMAND ===> SCROLL ===> CRSR MEMNAME BACKPO2 MEMLIB CTM.PROD.JOBLIB

OWNER M44 TASKTYPE JOB PREVENT-NCT2 Y DFLT N

APPL APPL-L GROUP BKP-PROD-L

DESC DAILY BACKUP OF SPECIAL FILES FROM APPL-L __OVERLIB CTM.OVER.JOBLIB STAT CAL __ SCHENV SYSTEM ID NJE NODE __ SET VAR CTB STEP AT NAME
DOCMEM BACKPO2 DOCLIB CTM.PROD.DOC TYPE DAYS AND/OR WDAYS MONTHS 1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- Y 11- Y 12- Y CONFCAL WORKDAYS SHIFT RETRO N MAXWAIT 04 D-CAT DEFINITION ACTIVE FROM UNTII ___ IN START-DAILY-BACKUP ODAT

Figure 24 Editing Job Scheduling Definitions in the Edit Environment Screen

A 2-character Line Editing command field, marked by underscores, is displayed for each line on the screen.

11.17.00

Editing commands are typed directly onto these underscores.

Specified Line Editing commands are processed when **Enter** is pressed.

Details and examples of the editing of job scheduling definitions in the Edit environment are provided in Appendix C, "Editing Job Scheduling Definitions in the Edit Environment."

## **Job Documentation**

### **Display and Non-Display of Documentation**

Depending on the value of the **Show Job Documentation** field in the scheduling facility entry panel, job documentation, in the form of DOC lines, is either displayed or hidden when you first enter the Job Scheduling Definition screen:

- If the **Show Job Documentation** field is set to Y, job documentation is displayed upon entry to the Job Scheduling Definition screen.
- If the **Show Job Documentation** field is set to N, documentation is hidden upon entry to the Job Scheduling Definition screen.

I

### **DOC Command**

Within a group, the DOC command toggles (alternately displays and hides) the group documentation. Within a job the DOC command toggles the job documentation.

### - NOTE -



Within a group, the DOC command only affects groups. Within jobs, the DOC command only affects jobs.

The DOC command remains in effect for the duration of an IOA online session.

Below is an example of the Job Scheduling Definition screen with the documentation (DOC lines) displayed.

Figure 25 Job Scheduling Definition DOC lines

```
JOB: BACKPLO2 LIB CTM.PROD.SCHEDULE
                                                              TABLE: BACKUP
COMMAND ===>
                                                              SCROLL ===> CRSR
  MEMNAME BACKPLO2 MEMLIB CTM.PROD.JOBLIB
  OWNER M44 TASKTYPE JOB PREVENT-NCT2 Y DFLT N
  APPL APPL-L
                                     GROUP BKP-PROD-L
  DESC DAILY BACKUP OF SPECIAL FILES FROM APPL-L
  OVERLIB CTM.OVER.JOBLIB
                                                              STAT CAL
  SCHENV
                                 SYSTEM ID
                                                              NJE NODE
  SET VAR
                      NAME
  CTB STEP AT
  CTB STEP AT NAME TYPE
DOCMEM BACKPLO2 DOCLIB CTM.PROD.DOC
                                      TYPE
  DOC THIS JOB BACKS UP SPECIAL "L" FILES. IT PERFORMS THE FOLLOWING STEPS:
  DOC 1: VERIFY SPACE REQUIREMENTS
  DOC 2-5: BACKUP THE FILES
  DOC 6: RECATALOG THE NEW FILES
  DOC 7: PRINT THE SHORT-VERSION LISTING REPORT
  DAYS ALL
                                                                     AND/OR
  WDAYS
                                                                 WCAL
COMMANDS: EDIT, DOC, PLAN, JOBSTAT
                                                                     11.17.00
```

#### NOTE -



If DOCU/TEXT is installed at your site, you can specify a DOCU/TEXT library and member with up to 132 characters per line.

## **Editing Documentation**

Documentation can be edited when the DOC lines of the Job Scheduling Definition screen are displayed. Modify the DOC lines as desired. When you fill in the last DOC line and press **Enter**, a new DOC line is displayed.

When modifying DOC lines, text must be left in at least one DOC line in order to save the modifications. Changes resulting in an empty DOCMEM member are not saved.

Job documentation is written to the library and member specified in the **DOCLIB** and **DOCMEM** fields on the Job Scheduling Definition screen. Therefore, it is also possible to edit the documentation member directly through ISPF. This is recommended when documentation is lengthy or the editing required is very complex.

### **Edit/Browse Mode Considerations**

In general, a documentation member is entered in edit mode unless any of the following is true, in which case the member will be entered in browse mode:

- The user has entered the table in browse mode (via option B).
- The documentation library has a record length greater than 80.
- There are non-blank characters in a documentation line beyond column 71. In addition, the displayed line is truncated.
- The documentation library is a Panvalet or Librarian library.
- There are non-displayable characters in the documentation member.
- Another user is viewing or editing the same documentation member. If profile variable SSCHBRO is set, see the chapter about IOA Administration in the *INCONTROL* for z/OS Administrator Guide for details.

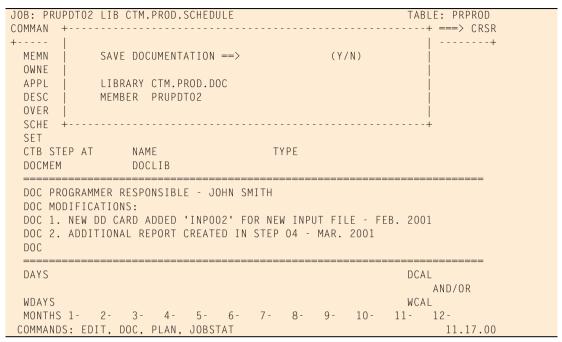
Security access authorization to the documentation library does not determine or limit the mode (edit/browse) in which the documentation member is entered (similar to ISPF member checking).

## **Auto-Save and Saving Documentation**

Documentation changes can be saved upon exiting the Job Scheduling Definition screen. When there are documentation changes, a Save Documentation window may be displayed depending on the value of the AUTO-SAVE DOCUMENTATION field in the Scheduling Facility entry panel:

- If the AUTO-SAVE DOCUMENTATION field was set to Y, documentation changes are automatically saved and the Save Documentation window is not displayed.
- If the AUTO-SAVE DOCUMENTATION field was set to N, documentation changes are not automatically saved and the Save Documentation window is displayed. This window lets you save or cancel the documentation changes.





The following parameters can be specified in the Save Documentation window:

**Table 43** Save Documentation Window Fields

Field	Description
SAVE DOCUMENTATION	Valid values are:
	<ul> <li>Y (Yes) – save documentation changes</li> <li>N (No) – do not save documentation changes</li> </ul>
LIBRARY	Name of the library containing the documentation member.
MEMBER	Name of the member containing the documentation.

Modify the LIBRARY and MEMBER values if desired, and type Y or N in the **SAVE DOCUMENTATION** field; then press **Enter**.

If there are no documentation changes, the Save Documentation window is not displayed.

# **Exiting the Scheduling Definition Facility**

When exiting the Scheduling Definition facility, screens are exited in the following sequence:

- Job Scheduling Definition screen
- Job List screen
- Table List screen

#### NOTE



If the Table List screen was bypassed when you entered the Scheduling Definition facility, that is, if you entered a TABLE value in the entry panel, the Table List screen is not displayed upon exiting the Job List screen; instead, the entry panel is displayed.

## **Entry Panel**

The commands and options available when exiting screens depend on the screen being exited and on whether changes have been made. If changes have been made, the selected exit options and commands determine whether the changes are saved. Exit options and commands are discussed below on a screen-by-screen basis.

## **Exiting the Job Scheduling Definition Screen**

Use any of the following commands, or press the corresponding PFKey, to exit the Job Scheduling Definition screen:

Table 44 Commands for Exiting the Job Definition Screen (part 1 of 2)

Command	Description				
CANCEL	Cancel the changes made to the job scheduling definition and return to the Job List screen.				
memory. To perm	<b>Note:</b> The following exit commands retain changes to the job scheduling definition in memory. To permanently save the changes to disk, you must request that the changes be saved when you exit the Job List screen:				
in "Auto-Save ■ For a job: If the panel, described displayed.	tion changes have been made, the Save Documentation window, described e and Saving Documentation" on page 146, may be displayed. The JOB parameter in a DO FORCEJOB request is blank, a confirmation and in "DO FORCEJOB: Post-Processing Parameter" on page 444, may be				
■ For a group: I	f the JOB parameter in a DO FORCEJOB request is blank, a confirmation				

END (**PF03/PF15**) Keep changes to the job scheduling definition in memory and exit to the Job List screen.

panel, described "ON GROUP-END: Post-Processing Parameter" on page 539, may be

displayed.

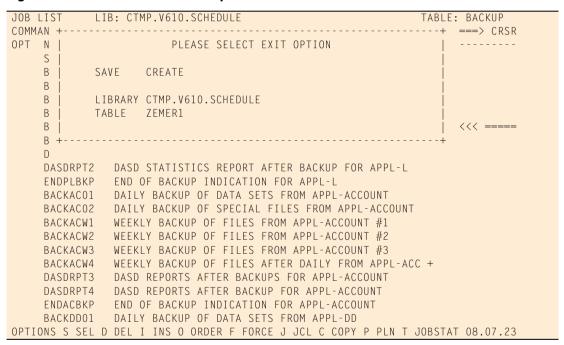
Table 44 Commands for Exiting the Job Definition Screen (part 2 of 2)

Command	Description
NEXT ( <b>PF11/PF23</b> )	Keep changes to the job scheduling definition in memory and display the next job scheduling definition from the Job list.
	Keep changes to the job scheduling definition in memory and display the previous job scheduling definition from the Job list.

## **Exiting the Job List Screen**

Press END (**PF03/PF15**) to exit the Job List screen. If changes made to at least one job scheduling definition have been kept in memory, as discussed in the preceding section, "Exiting the Job Scheduling Definition Screen", and/or if changes have been made to the Job List screen, the Exit Option window is displayed.

Figure 27 Job List Screen Exit Option Window



The LIBRARY and TABLE fields indicate the library and table in which the job scheduling definitions will be saved. The specified values can be modified, for example, to save the job scheduling definitions in a new or different table.

Fill in the Exit Option window as follows:

- To save all changes currently in memory and exit the Job List screen, type Y (Yes) after the word SAVE or CREATE:
  - Type Y after SAVE if a table with the same name already exists in the specified library.
  - Type Y after CREATE if a table with the same name does not exist in the specified library.

### - NOTE -



If you create a new table, the table name does not appear in the Table List screen upon exiting the Job List screen; it first appears when you reenter the Table List screen from the entry panel.

- To cancel changes currently in memory and exit the Job List screen, type N (No) after the word SAVE or CREATE.
- To close the Exit Option window and remain in the Job List screen, with the changes remaining in memory, press RESET (PF04/PF16).

## **Exiting the Table List Screen**

Press the END key (PF03/PF15) to exit the Table List screen.

## **Exiting the Entry Panel**

Press the END key (PF03/PF15) to exit the entry panel.

# **Ordering (Scheduling) Jobs**

Although job scheduling in the production environment is generally handled by CONTROL-M automatically (for more information, see Chapter 6, "Selected Implementation Issues"), CONTROL-M provides several mechanisms for scheduling jobs manually. Among these are options to manually request job scheduling from the Table List screen and the Job List screen.

■ When manually requesting job scheduling from the Job List screen, specific jobs are selected. Multiple jobs can be specified.

■ When manually requesting job scheduling from the Table List screen, tables are selected and each request applies to all the jobs in the selected tables. Multiple tables can be specified.

Either of two options, O (Order) and F (Force), can be used in either of these screens to manually request job scheduling. These options work as follows:

**Table 45 Options for Manually Ordering Jobs** 

Option	Description
O (ORDER)	Basic scheduling parameters of the jobs are checked against the requested scheduling date. If the job must be scheduled for that day, a job order is placed on the Active Jobs file.
F (FORCE)	Basic scheduling parameters of the jobs are not checked. A job order is placed on the Active Jobs file that day even if the scheduling criteria of the job are not satisfied.

### NOTE



With one exception, options O (Order) and F (Force) can be used in both the Job List screen and Table List screen for either regular scheduling tables or Group scheduling tables.

The exception is: Option O (Order) cannot be entered for individual jobs in Group scheduling tables.

When you use the O and F options, a confirmation window is opened. The default confirmation window in the case where the O option has been entered for schedule tables in the Table List screen is illustrated in Figure 28.

Figure 28 Order and Force Confirmation Window (Groups)

```
LIST OF TABLES IN CTM.PROD.SCHEDULE
                                                                TABLE: BACKUP
COMMAND ===>
                                                                SCROLL ===> CRSR
OPT NAME ---
      STARTBKP
      BACKPLO1 <----- CONFIRM ODATE 210205 WAIT FOR ODATE N HOLD |
      BACKPL02
                          +-----
      BACKPIW1
      BACKPLW2 WEEKLY BACKUP OF FILES FROM APPL-L #2
      BACKPLW3 WEEKLY BACKUP OF FILES FROM APPL-L #3
      BACKPLW4 WEEKLY BACKUP OF FILES AFTER DAILY FROM APPL-L +
      DASDRPT1
                DASD REPORTS AFTER BACKUPS FOR APPL-L
      DASDRPT2 DASD STATISTICS REPORT AFTER BACKUP ENDPLBKP END OF BACKUP INDICATION FOR APPL-L BACKACO1 DAILY BACKUP OF DATA SETS FrROM APP
                DASD STATISTICS REPORT AFTER BACKUP FOR APPL-L
                DAILY BACKUP OF DATA SETS FRROM APPL-ACCOUNT
     BACKACO2 DAILY BACKUP OF SPECIAL FILES FROM APPL-ACCOUNT
      BACKACW1 WEEKLY BACKUP OF FILES FROM APPL-ACCOUNT #1
      BACKACW2 WEEKLY BACKUP OF FILES FROM APPL-ACCOUNT #2
      BACKACW3 WEEKLY BACKUP OF FILES FROM APPL-ACCOUNT #3
      BACKACW4 WEEKLY BACKUP OF FILES AFTER DAILY FROM APPL-ACC +
      DASDRPT3 DASD REPORTS AFTER BACKUPS FOR APPL-ACCOUNT
      DASDRPT4 DASD REPORTS AFTER BACKUP FOR APPL-ACCOUNT
      ENDACBKP END OF BACKUP INDICATION FOR APPL-ACCOUNT
      BACKDD01 DAILY BACKUP OF DATA SETS FROM APPL-DD
OPTIONS S SELECT O ORDER F FORCE G GRAPHIC FLOW B BROWSE D DELETE 10.23.58
```

If the O or the F option is entered for jobs in the Job List screen, a window similar to that in Figure 29 appears.

Figure 29 Order and Force Confirmation Window (Jobs)

```
LIB: CTM.PROD.SCHEDULE
JOB LIST
                                                           TABLE: BACKUP
COMMAND ===>
                                                           SCROLL ===> CRSR
OPT NAME -----
     STARTBKP
               | CONFIRM y ODATE 2210003 WAIT FOR ODATE N
     BACKPLO2 <--- | Dynamic insert job into group s DUPLICATE Y
     BACKPLW1 | (S - Select, R - Recent, N - New, A-alone - default
     BACKPLW2 WEE+-----
     BACKPLW3 WEEKLY BACKUP OF FILES FROM APPL-L #3
     BACKPLW4 WEEKLY BACKUP OF FILES AFTER DAILY FROM APPL-L +
     DASDRPT1
               DASD REPORTS AFTER BACKUPS FOR APPL-L
     DASDRPT2
               DASD STATISTICS REPORT AFTER BACKUP FOR APPL-L
     ENDPLBKP
               END OF BACKUP INDICATION FOR APPL-L
     BACKAC01
               DAILY BACKUP OF DATA SETS FROM APPL-ACCOUNT
     BACKAC02
               DAILY BACKUP OF SPECIAL FILES FROM APPL-ACCOUNT
     BACKACW1
               WEEKLY BACKUP OF FILES FROM APPL-ACCOUNT #1
     BACKACW2 WEEKLY BACKUP OF FILES FROM APPL-ACCOUNT #2
     BACKACW3 WEEKLY BACKUP OF FILES FROM APPL-ACCOUNT #3
     BACKACW4 WEEKLY BACKUP OF FILES AFTER DAILY FROM APPL-ACC +
     DASDRPT3 DASD REPORTS AFTER BACKUPS FOR APPL-ACCOUNT
     DASDRPT4 DASD REPORTS AFTER BACKUP FOR APPL-ACCOUNT
     ENDACBKP END OF BACKUP INDICATION FOR APPL-ACCOUNT
     BACKDD01 DAILY BACKUP OF DATA SETS FROM APPL-DD
OPTIONS S SEL D DEL I INS O ORDER F FORCE J JCL C COPY P PLN T JOBSTAT 15.37.39
```

The default confirmation window contains the following fields:

Table 46 Order and Force Confirmation Window Fields (part 1 of 2)

Field	Description						
CONFIRM	Whether to process the Order or Force request.						
	Valid values are:						
	<ul> <li>Y (Yes) - Process the Order or Force request.</li> <li>N (No) - Cancel the request.</li> </ul>						
ODATE	Scheduling date of the job or table, in mmddyy, ddmmyy or yymmdd format, depending on the site standard. The specified date can be modified.						
	<b>Note:</b> The job is only ordered if the basic job scheduling criteria are satisfied at the ODATE. However, the job is not necessarily executed on the ODATE. If the job is ordered, it becomes eligible for execution immediately when its run-time criteria have been satisfied.						
	ODATE has the following additional functions:						
	■ If an IN or OUT condition uses a relative <i>dateref</i> (for example if the value of <i>dateref</i> is ODAT, PREV, or NEXT), ODATE is used to set the <i>dateref</i> . For more information on IN and OUT conditions, see "IN: Runtime Scheduling Parameter" on page 500 and "OUT: Post–Processing Parameter" on page 562.						
	■ The calculation of the MAXWAIT of a job is based on the ODATE of the job, and not on the actual date on which the job is ordered. For more information on the MAXWAIT parameter, see "MAXWAIT: Basic Scheduling Parameter" on page 519.						
	It is used by the WAIT FOR ODATE option (as described in this table).						
	For more information on the meaning of ODATE, see the discussion of Time Zone support in the CONTROL-M chapter of the <i>INCONTROL for z/OS Administrator Guide</i> .						
WAIT FOR ODATE	Whether to wait for a specific date, or process the Order or Force request immediately.						
	Valid values are:						
	■ Y (Yes) – The Order or Force request is not executed before the date set in the <b>ODATE</b> field, even if all required conditions and resources are available.						
	■ N (No) – The Order or Force request is processed immediately. Default.						

Table 46 Order and Force Confirmation Window Fields (part 2 of 2)

Field	Description
HOLD	Whether or not jobs should be held immediately after being ordered.
	Valid values are:
	<ul> <li>Y - Jobs will be held immediately after being ordered.</li> <li>N - Jobs will not be held. Default.</li> </ul>
Dynamic insert job into group	Enables you to insert a job into a group. Valid values are:
into group	<ul> <li>A - Order or FORCE the job. Default.</li> <li>S - Opens a window with a list of groups. For more information, refer to "Selecting Dynamic Jobs" on page 155.</li> <li>R - Adds the job to the group entity that was most recently ordered into the AJF. The insertion to the AJF is done as FORCE.</li> <li>N - Force a job, and its group, to the AJF.</li> </ul>
	Notes:
	<ul> <li>If an attempt is made to use option "R" (recent) to add a job to a deleted group, error message JOB53BE is displayed. Also, if an attempt is made to use option "R" (recent) to add multiple jobs (ASK FOR EACH ONE=N), only the first job is ordered.</li> <li>If the "G" (group) option is entered for the group, the added job is displayed after the group entry and not at the end of the list.</li> </ul>
DUPLICATE	Determines whether you can dynamically insert duplicate jobs (meaning, jobs that already exist in the group entity of the AJF). Valid values are:
	<ul> <li>Y - A duplicate job can be inserted. Default</li> <li>N - A duplicate job can not be inserted. Attempting to insert a duplicate job results in the display of an error message</li> </ul>
	This parameter is only valid if the Dynamic insert job into group parameter is set to S or R.
ASK FOR EACH ONE	This line is displayed only if more than one order or force option is requested. It determines whether individual confirmation is required for each order or force request.
	Valid values are:
	■ Y (Yes) – Individual confirmation is required for each order or force request. The specified CONFIRM value (Y or N) applies only to the current order or force request.
	■ N (No) – Individual confirmation is not required for each order or force request. The specified CONFIRM operation is applied to all order or force requests. If CONFIRM is Y, all order or force requests are processed; if CONFIRM is N, no order or force requests are processed.

Fill in the confirmation window and press **Enter**. If at least one order or force request has been specified for a table or job, the original list screen disappears and a message screen is displayed. This screen displays messages that contain the following information about the jobs that were scheduled.

```
JOB name ODATE date ID=ordered PLACED ON ACTIVE JOBS FILE-descr
```

Each iteration of the message screen displays job information for one table only. Press END (PF03/PF15) to exit the message display for that table. If multiple tables, or jobs in multiple tables, have been scheduled, the messages for the next table are displayed. When messages for all tables have been displayed, pressing END displays the Table or Job list screen.

## **Selecting Dynamic Jobs**

Entering the value S in the Dynamic insert job into group parameter opens the Select Group window (Figure 30).

Figure 30 Select Group window

```
Filter:
                   ----- CONTROL-M Dynamic Ins Job/Grp ----- UP <D> - (3)
COMMAND ===>
                                                      SCROLL ===> CRSR
         O NAME
  CONTROLM 301104
                                        GRP Held (Ordering) Group=OPHIR3
 MEM1
                                           OrderID=000FY
                                        GRP Active Group=OPHIR3
 MEM1
         CONTROLM 291104
                                           OrderID=00007
                                        GRP Held (Ordering)
 GRPADJUS K30
                 161204
                                          Group=GRP_WM3012 OrderID=000UN
 GRPADJUS K30
                 161204
                                        GRP Held (Ordering)
                                          Group=GRP_WM3012 OrderID=000UQ
S - Select this group
```

To select the group entity into which you force the job into the AJF, enter S on the relevant line.

#### - NOTE



All other commands in this screen are invalid. Note that the status of the group may change as a result of adding the job to the group.

## **The Double Confirmation Window**

Any request to order or force a job can, if you prefer, only be processed if it has been confirmed twice. This option is selected by means of the SSCHTBO parameter.

The SSCHTBO parameter is one of the parameters set in the Presentation Modes minor step in the Profile Variables major step of the Customize option (Option 6) of the INCONTROL Installation and Customization Engine (ICE). The default setting is N (No), meaning that when a table is ordered or forced, a regular confirmation window is opened, and not a double confirmation window.

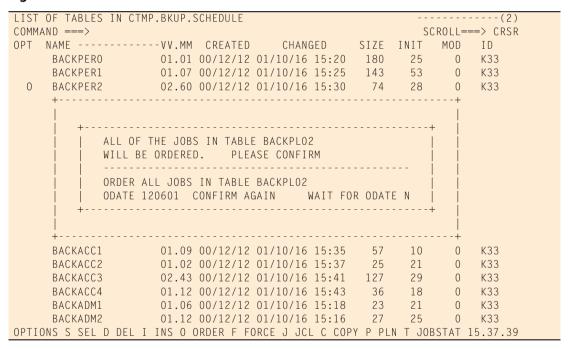
The SSCHTBO parameter can be modified at any time. If the setting of the SSCHTBO parameter is modified to Y (Yes), a window that requires double confirmation is opened in response to any order or force request. An example of this window is shown in Figure 31.

#### — NOTE -



If you change the setting of the parameter, you must exit and then reenter the IOA online environment before the change can take effect.

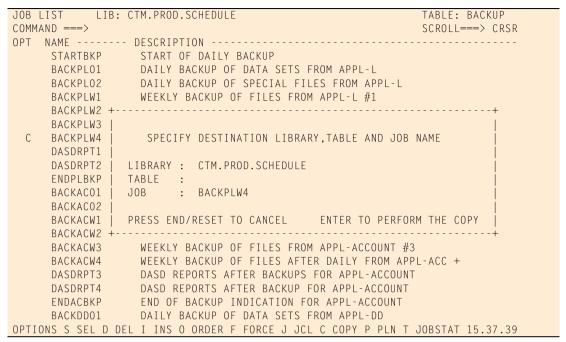
Figure 31 The Double Confirmation Window



# **Copying Jobs to Another Table**

To copy one or more jobs from the current table to another table, type **C** (Copy) in the OPT column of the Job List screen, next to the job names, and press **Enter**. The following window is displayed:

Figure 32 Window for Copying Jobs to Another Table



The window contains the fields shown in the following table. Some fields contain default values that can be modified.

Table 47 Fields in the Window for Copying Jobs to Another Table

Field	Description
Library	Library containing the table into which the jobs must be copied. Must be an existing library. Default is the current library.
Table	Name of the table into which the job must be copied.
	<b>Notes:</b> A job can only be copied to another table. It cannot be copied to its own table (even if the job is renamed).
	If the specified table does not exist, the table is created when the request is performed.
Job	Name of the job to be copied. If multiple jobs are selected, the window initially display with the first selected job. As each request is performed or canceled, the next requested job name is displayed.

To perform a request, press **Enter**.

To cancel a request, press END (PF03/PF15) or RESET (PF04/PF16).

Group entities cannot be copied. If a job in a Group scheduling table is copied to a regular scheduling table, it is copied as a regular job; scheduling tags are dropped from the job scheduling definition. If a job in a Group scheduling table is copied to a nonexisting table, the table that is created is a regular table, not a group table.

#### - NOTE



When a job from a regular table is copied to a Group scheduling table, the job retains its basic scheduling criteria and an empty **SCHEDULE TAG** field is inserted in the job scheduling definition.

When a job from a Group scheduling table is copied to a Group scheduling table, the job retains its original scheduling tags.

# **Deleting Tables**

Tables can be deleted from the Table List screen.

To delete tables, type D (Delete) by the table names in the Table List screen and press **Enter**.

The confirmation window illustrated below is displayed, in sequence, for each table selected for deletion.

Figure 33 Delete Table Confirmation Window

LIST	OF TABLES I	IN CTM.PROD.SCI	HEDULE			(2)	
COMMA	AND ===>				SCROLL===> CRSR		
OPT	NAME	-	+	-+ E	INIT	MOD ID	
	ADABAS		CONFIRM DELETE OPTION	0	70	0 001	
D	APPLNTN	<	(Y/N)	0	180	0 001	
	APPLPRDI	-	+	-+ 1	41	0 001	
	ARCNIGHT	01.00	01/02/09 01/06/07 00:50		5	0 S07	
	ASMBTR1	01.00	01/02/09 01/06/07 00:50	9	9	0 S07	
D	ASMBTR2	01.00	01,02,03 01,00,0, 00.00	14	14	0 S07	
	BACKUP	01.00	01/02/09 01/06/07 00:50	5	5	0 S07	
	CICSJOBS	01.00	01/02/09 01/06/07 00:50	70	70	0 001	
	CICSPROD	01.00	01/02/09 01/06/07 00:50	180	180	0 001	
	CICSTEST	01.00	01/02/09 01/06/07 00:50	41	41	0 001	
	CICSUPT	01.00		5	5	0 S07	
	CLIENTS	01.00	01/02/09 01/06/07 00:50	9	9	0 S07	
	DB2EXE	01.00	01/02/09 01/06/07 00:50	14	14	0 S07	
	DLOAD	01.00	01/02/09 01/06/07 00:50	5	5	0 S07	
	MAINDAY	01.00	01/02/09 01/06/07 00:50	180	180	0 001	
	MAINT	01.00	01/02/09 01/06/07 00:50	41	41	0 001	
	MAINTPL	01.00		5	5	0 S07	
	ONSPOOL	01.00	01/02/09 01/06/07 00:50	9	9	0 S07	
D	ONSPOOLT	01.00	01/02/09 01/06/07 00:50	14	14	0 S07	
	OPERCLN	01.00	01/02/09 01/06/07 00:50	5	5	0 S07	
OPTIO	ONS S SELECT	Γ O ORDER F I	FORCE G GRAPHIC FLOW B B	ROWSE	D DEL	ETE 15.38.37	

Type Y (Yes) in the window to confirm the delete request.

Type N (No) in the window to cancel the delete request.

A message is written to the IOA Log file for each table deleted.

#### — NOTE



If PDSMAN is operational at your site, \$\$\$PACE members are not deleted.

## **Condition Inheritance**

When a job is deleted, there is an optional feature in CONTROL-M that automatically adjusts the jobs in the table so that all the jobs that are waiting for a deleted job's OUT conditions will inherit its IN conditions.

When a job that is part of a table is deleted, and profile variable SSCHCDJ=Y, the following window is displayed:

Figure 34 Condition Inheritance Confirmation Window

JOB LIST LIB: CTMP.V630.SCHEDULE	TABLE: CICS
COMMAND ===>	SCROLL===> CRSR
OPT NAME DESCRIPTION	
CICSPRD1 CICSPRD1	
CICSPRD2 CICSPRD2	
CICSTST1 CICSTST1	
CICSTST2 CICSTST2	
SYSTST1 SYSTST1	
SYSTST2 +	·
The state of the s	(/N)
PROD2	( ( ) )
PROD3   AUTO INHERIT JOBS IN-CONDITIONS N ()	(/N)
USER1 +	+
USER2 USER2 USER3 USER3	
SYSPROD1 SYSPROD1	
SYSPROD2 SYSPROD2	
SYSPROD3 SYSPROD3	
LOG1 LOG1	
LOG2 LOG2	
LOG3 LOG3	
LOG4 LOG4	
LOG5 LOG5	
OPTIONS S SEL D DEL I INS O ORDER F FORCE J JCL C COPY P PLN	T JOBSTAT 08.34.31

The default value of the confirm delete option is blank.

The default value of the auto inherit job's in-conditions option is N.

When the confirm delete option is blank or N, the job is not deleted.

When the confirm delete option is Y and the auto inherit job's in-conditions option is N, the job is deleted as usual.

When the confirm delete option is Y and the auto inherit job's in-conditions option is Y, all the jobs that are waiting for the deleted job's OUT conditions will inherit its IN conditions.

A message is written to the IOA log file for each job deleted.

### NOTE



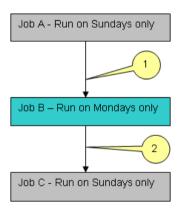
If PDSMAN is operational at your site, \$\$\$SPACE members are not deleted.

## **Limitations**

Condition inheritance applies to one table only. Relationships between tables are ignored.

### **Unusual job definition 1**

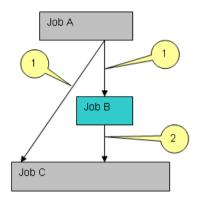
This enhancement does not handle "unusual" job definitions. For example, assume that three jobs are present in a table:



Under normal conditions, jobs B and C would never run. However, if job B is deleted using the condition inheritance feature, job C will incorrectly run every Sunday.

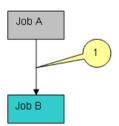
### **Unusual job definition 2**

If job B deletes condition 1, job C will never run. However, if job B is deleted using the condition inheritance feature, job C will incorrectly run.



### **Unusual job definition 3**

Assume that job B deletes output condition X. If job B is deleted using the condition inheritance feature, the deletion of output condition X will be lost.



### Other feature limitations

Parentheses are rearranged

In order to use the condition inheritance feature, CONTROL-M formats must be converted to standard Boolean format, manipulated, and then converted back to CONTROL-M format. In this case, the parentheses are rearranged, although the functionality remains the same. For example:

CONTROL-M format: A(|B|C)

becomes

CONTROL-M format after manipulations: | (A B) | (A C)

Adjust condition is ignored

The Adjust Condition feature for a group table is ignored.

■ DO COND is ignored

Output conditions that are associated with a DO COND statement are ignored.

Manual conditions

Manual conditions are treated the same way as regular conditions. This may lead to incorrect results.

# **Displaying Graphic Jobflow**

The Graphic Jobflow screen provides a graphic presentation of the job flow dependencies in a given scheduling table. It is displayed when option G (Graphic Flow) is specified for a table in the Table List screen.

Use the shifting PFKeys to shift the Graphic Jobflow right (PF11/PF23) and left (PF10/PF22).

The FIND command (PF05/PF17), described in "FIND Command" on page 96, is supported in the Graphic Jobflow screen.

To return to the Table list, press the END key (PF03/PF15).

Two formats for the Graphic Jobflow screen are available, one for color displays and one for non-color displays.

# **Graphic Jobflow for Color Terminals**

Figure 35 Graphic Jobflow for Color Terminals

```
---> PROJYDPY ---> PROJYDTK ---> PROYH11 ---
---> PROJYDLI --->
PRESS END TO RETURN. LEFT AND RIGHT TO SEE MORE. COLUMNS: 001 - 080 15.38.44
```

### NOTE



Size limits for the online display are narrower than the size limits for the printed chart. The online display is limited to 15 levels of dependencies. If the chart cannot be displayed online because it is too large, it can still be printed. For more information, see the description of the CTMRFLW utility in the CONTROL-M chapter of the *INCONTROL for z/OS Utilities Guide*.

On color terminals, the colors used to display the boxes and arrows can be changed by request. Available colors are sequenced in a loop. Each request changes to the next color in the sequence. Colors can be changed as follows:

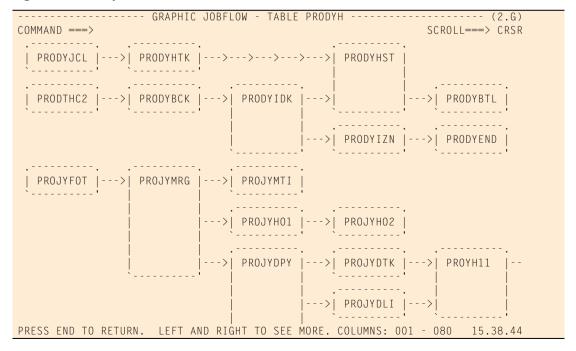
**Table 48 Color Change Options on Graphic Jobflow Window** 

Option	Description
Change the color of the boxes	Press PF04/PF16
Change the color of arrows	Type ARROW in the <b>COMMAND</b> field and press <b>Enter</b> . ^a

PF05/PF17, which used to change the color of arrows, now performs the FIND command.

## **Graphic Jobflow for Non-Color Terminals**

Figure 36 Graphic Jobflow for Non-Color Terminals



### - NOTE -



Size limits for the online display are narrower than the size limits for the printed chart. If the chart cannot be displayed online because it is too large, it may still be printed. For more information, see the *KeyStroke Language (KSL) User Guide*.

# **Displaying a Job Scheduling Plan**

To display the scheduling plan of a job or group entity in calendar format, either type P (Plan) by the job or group entity name in the Job List screen and press **Enter**, or enter command PLAN in the Job Scheduling Definition screen. The following window is displayed:

Figure 37 Job Scheduling Plan Window

```
JOB LIST
            LIB: CTM.PROD.SCHEDULE
                                                           TABLE: BACKUP
COMMAND ===>
                                                           SCROLL===> CRSR
OPT NAME ---- DESCRIPTION ------
              START OF DAILY BACKUP
     STARTBKP
     BACKPL01
               DAILY BACKUP OF DATA SETS FROM APPL-L
     BACKPLO2 DAILY BACKUP OF SPECIAL FILES FROM APPL-L
     BACKPLW1 WEEKLY BACKUP OF FILES FROM APPL-L #1
     BACKPLW2 WEEKLY BACKUP OF FILES FROM APPL-L #2
     BACKPLW3 WEEKLY BACKUP OF FILES FROM APPL-L #3
     BACKPLW4 WEEKLY BACKUP OF FILES AFTER DAILY FROM APPL-L +
     DASDRPT1 DASD REPORTS AFTER BACKUPS FOR APPL-L
     DASDRPT2 DASD STATISTICS REPORT AFTER BACKUP FOR APPL-L
     ENDPLBKP
                           +----+
     BACKAC01
                             FROM DATE 050101
     BACKAC02
                             TO DATE 053101
     BACKACW1
     BACKACW2
     BACKACW3 WEEKLY BACKUP OF FILES FROM APPL-ACCOUNT #3
     BACKACW4 WEEKLY BACKUP OF FILES AFTER DAILY FROM APPL-ACC +
     DASDRPT3 DASD REPORTS AFTER BACKUPS FOR APPL-ACCOUNT
     DASDRPT4 DASD REPORTS AFTER BACKUP FOR APPL-ACCOUNT
     ENDACBKP END OF BACKUP INDICATION FOR APPL-ACCOUNT
     BACKDD01 DAILY BACKUP OF DATA SETS FROM APPL-DD
OPTIONS S SEL D DEL I INS O ORDER F FORCE J JCL C COPY P PLN T JOBSTAT 15.37.39
```

The window contains the following fields with default values that can be modified:

Table 49 Job Scheduling Plan Window Fields

Field	Description
FROM DATE	First date to be included in the job scheduling plan. Default is current working date.
TO DATE	Last date to be included in the job scheduling plan. Default is one day less than one month after the current working date.

To display the plan, modify the defaults if desired. Press **Enter**. The Job Scheduling Plan screen is displayed.

To close the window without displaying the plan, press END (PF03/PF15) or RESET (PF04/PF16).

## **Job Scheduling Plan Screen**

For each month in the requested date range, beginning with the first month in the range, the Job Scheduling Plan screen displays a calendar that indicates the schedule of the job.

### NOTE



The months (within the date range) in which the job (or jobs for a group schedule) is not scheduled are not displayed.

The dates within the date range on which the job is scheduled, according to its basic scheduling criteria, are marked with an asterisk.

Enter NEXT (PF08/PF11/PF20/PF23) or PREV (PF07/PF10/PF19/PF22) in the COMMAND field, or press the appropriate PFKey, to see the next or previous calendar month in the date range.

Press the END key (**PF03/PF15**) to exit the Job Scheduling Plan screen and return to the Job List screen.

Figure 38 Job Scheduling Plan Screen

JOB NAME: BACKPLO2 COMMAND ===>			JOB	SCHEDU	JLING		DATES	: 010601 - 300601 SCROLL===>
06 2001	MON	TUE	WED	THU	FRI 01	SAT 02	SUN 03	JONOLL /
	04	05	06	07	08	09	10	
	*	*	*	*	*			
	11	12	13	14	15 *	16	17	
	18	19	20	21	22	23	24	
	25 *	26	27	28	29	30		
	*	*	*	*	*			
CMDS: NEXT, PREV,	END							19.30.51

The screen indicates the following:

Table 50 Job Scheduling Plan Screen Fields

Field	Description
Job	Name of the job for which the schedule is requested.
Range	Requested date range, in mmddyy, ddmmyy, or yymmdd format, depending on the site standard.
Month/Year	Month and Year currently displayed.
Calendar	Calendar (day of the week and date) for the currently displayed month.
Schedule	An asterisk under a date indicates that the job is scheduled on that day.

# **Tracking and Control Facility**

The Tracking and Control facility provides relevant information about the status of each job and task in the Active environment and enables you to manually intervene in the processing of jobs.

The Active environment contains all the jobs in the Active Jobs file, that is, all jobs that have recently executed, are currently executing, or are scheduled for possible execution in the near future.

The main screen of the Tracking and Control facility is the Active Environment screen, which displays a list of all jobs and their statuses in the Active environment.

The Active Environment screen is accessed by requesting option 3 on the IOA Primary Option menu.

From the Active Environment screen you can request specific actions in relation to a job, or request the display of other screens that provide additional information and capabilities. Possible requests include

- Change the screen display type to display different information about the jobs in the Active environment.
- View only jobs belonging to a specific group.
- View statistical information about all jobs in the Active environment.
- Display the reasons why a job has not executed.
- Place a job in Hold status.
- Delete a job from the Active environment.

- Delete a Group entity and all its jobs from the Active environment.
- Undelete a job that has been deleted from the Active environment.
- Free a held job.
- Display the log messages of a job.
- Zoom in on the scheduling details of a job and modify certain parameters.
- Rerun a job.
- Confirm the restart and/or rerun of a job.
- View the list of all runs (job orders) of a particular job, and view the sysout of any or all of those job orders.
- Display the execution statistics of a job.
- Display and edit the JCL of a job.
- Display the list or network of predecessor and successor jobs of a specific job, and display critical path information.
- Force END OK termination of a job that has not been submitted or that ended NOTOK.
- Reactivate a job that has "disappeared" or that has failed with status "Reason Unknown".
- Display the list of archived jobs (in the History file) and restore desired archived jobs to the Active environment.

## **Active Environment Screen**

To enter the Tracking and Control facility, select option 3 on the IOA Primary Option menu. The Active Environment screen is displayed.

Figure 39 CONTROL-M Active Environment Screen

```
Filter:
                     ----- CONTROL-M Active Environment ----- UP
                                                                                         (3)
COMMAND ===>
                                                                           SCROLL ==> CRSR
O Name
            Owner
                       Odate Johname JobID Typ ------ Status -----
  CICSPROD M22
                       060601 CICSPROD/04368 CST Executing (Run 1)
  CICSTEST M22
                       060601 CICSTEST/04372 CST Executing (Run 2)
  BRIVPCC IVP
BRCC0001 IVP
BRCC0002 IVP
                       060601
                                                    GRP Active
                       060601 BRCC0001/04382
                                                    JOB Held Wait Schedule
  BRCC0002 IVP
                       060601 BRCC0002/04383
                                                    JOB Held Ended "OK" (Run3)
                                                        Prior Run: Ended "OK"
  BRCCIND IVP
BRUPDTO2 IVP
BRREPO02 IVP
BRIVPCCE IVP
CRCCEND IVP
INTRO001 M22
                       060601 BRCCIND /04385
                                                    JOB Ended "OK"
                       060601 BRUPDT02/04387
                                                   JOB Ended "OK'
                       060601 BRREP002/04389
                                                   JOB Ended "OK"
                                        /__/
                       060601
                                                    JOB Wait Schedule
                       060601
                                                   JOB Wait Schedule
                                                   JOB Ended- Not "OK" Due to CC -
                       060601
                                                         Rerun Needed (Run 3)
                                                        Prior Run: Ended- Not "OK" Due
                                                        to CC - Rerun was Needed
  INTR0003 M22 060601 / JOB Wait Schedule
BRREP003 IVP 060601 BRREP003/04391 JOB Ended "OK"
BRREP004 IVP 060601 BRREP004/04393 JOB Ended "OK"
INTR0004 M22 060601 INTR0004/04397 JOB Ended Not "OK" - Abended
Commands: OPt DIsplay Show HIstory RBal REFresh Auto Jobstat SHPF Note Table
            OPt command toggles between Commands and Options display 15.15.48
```

It is assumed that you want to see the most recently ordered jobs first. Therefore, by default, the bottom of the Job list is displayed upon entry to the Active Environment screen. This default can be altered by setting Profile variable SACTMOD to T, in which case jobs are displayed from the top of the Job list. Profile variable SACTMOD is described in the *INCONTROL* for *z/OS* Administrator Guide.

AutoRefresh mode is available in this screen.

To exit the Active Environment screen, press the END key (PF03/PF15).

For color terminals, jobs with different statuses are displayed in different colors. Each of the colors in the following table has been defined as the default for one of these statuses. These statuses are described more fully in Table 64 on page 193.

#### — NOTE -



To change color definitions, see your INCONTROL administrator.

Table 51 Default Colors for Active Missions Screen (part 1 of 2)

Color	Corresponding Status
Blue and White	Jobs waiting to be scheduled
Yellow	Jobs that are executing or about to execute
Red	Jobs that are in error or ended NOTOK or LATE (submitted and/or executing jobs)

Table 51 Default Colors for Active Missions Screen (part 2 of 2)

Color	Corresponding Status
Green	Jobs that ended OK or were forced OK
Pink	Jobs that require special user action (such as Wait Confirmation)

# **Display Types of the Active Environment Screen**

The information in the Active Environment screen can be displayed in different formats or display types. A number of predefined display types are available.

While in the Active Environment screen, the display type can be changed by the DISPLAY command. The DISPLAY command is described in "Commands of the Active Environment Screen" on page 174.

**Table 52** Predefined Display Types

Туре	Description
D	Default display type.
A	All info display type.
N	Network display type.

The INCONTROL administrator can use display type support to tailor the display layout by adding lines, fields, changing colors, and so on. Additional information about display type support is provided in the section on customizing IOA display format members in the *INCONTROL* for *z/OS* Administrator Guide.

### - NOTE



The \$\$ACTDOC member in the IOA MSG library contains information that is useful for customizing the CONTROL-M Active Environment screen and creating and modifying display types for screens 3, 3.N, 3.G and the History Environment screen.

Upon reentering the screen, the last-used display type is displayed.

The Default and All Info predefined display types, and the fields they contain, are discussed in "Format of the Active Environment Screen" on page 170.

The Network display type, although available in this screen, is generally useful only in the Job Dependency Network screen, and is described in "Job Dependency Network Screen" on page 242.

### **Primary and Alternate Bottom Lines**

The last two lines of the Active Environment screen are used to display the list of available commands and options.

Because there are too many commands and options to list at once, the list is divided into two parts, each part consisting of two lines, as follows:

■ Upon entry to the screen, the set of available commands is displayed. Because this list is displayed upon entry to the screen, it is referred to as the Primary Bottom line.

```
Commands: OPt DIsplay Show HIstory RBal REFresh Auto Jobstat SHPF Note Table OPt command toggles between Commands and Options display 15.15.48
```

■ Upon request (using command OPT), the set of available options is displayed (in place of the set of commands). The list of available options is referred to as the Alternate Bottom line.

```
Opt: ? Why L Log H Hold Z Zoom R Rerun A Activate O Force OK V View Sysout N Net D Del F Free S Stat G Group U Undelete J JCL Edit C Confirm 15.46.06
```

To toggle back and forth between the two sets of bottom lines, type **OPT** in the **COMMAND** field and press **Enter**.

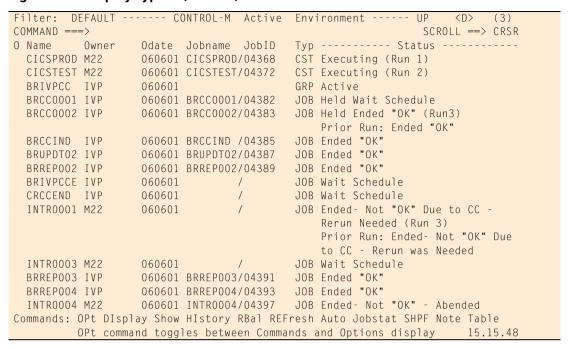
Available commands are described in "Commands of the Active Environment Screen" on page 174. Available options are described in "Options of the Active Environment Screen" on page 180.

## Format of the Active Environment Screen

### **Display Type D (Default)**

Below is an example of the Default display type. It contains the most relevant information about the job.

Figure 40 Display Type D (Default)



### Fields of Display Type D (Default)

Table 53 Fields in the Default Display Type

Field	Description
Filter	Name of the currently active screen filter. For more information, see "Filtering the Active Environment Screen Display" on page 189.
CONTROL-M Status	Indicator of whether CONTROL-M monitor is UP, DOWN or SUSP (suspended).
Display Type	Indicator of the currently used display type, such as D for the Default display type.

The following is displayed for each job:

Table 54 Fields for Each Job Default (part 1 of 2)

Field	Description
O(ption)	Field for requesting options to be activated on jobs.
Name	Name of the member containing the JCL of the job, or name of the started task.
Owner	ID of the owner of the job.
Odate	Original scheduling date of the job. For more information, see "Date Definition Concepts" on page 63.
Jobname	Job name – generally available only after job submission.
JobID	Job number – generally available only after job submission.

Table 54 Fields for Each Job Default (part 2 of 2)

Field	Description
Тур	Task type or GRP.
Status	Job (task) status, For more information, see "Job Statuses" on page 185.

The information in the following table can be displayed by request, along with the STATUS information. For more information, see "Commands of the Active Environment Screen" on page 174.

**Table 55** Other Information in the STATUS Field

Туре	Descriptions
GROUP	Group name of the job, discussed in Chapter 3, "Job Production Parameters."
CPU	ID of the CPU on which the job is executing or has executed. The CPU ID is displayed only for those jobs subjected to dynamic resource acquisition (that is, jobs for which CONTROL-M dynamically decided to which CPU they would be submitted). Dynamically acquired resources are identified by a \$ character in the Quantitative resource name. For more information, see "CPUID" on page 178.
OrderID	Order ID of the job. For more information, see "ORDERID" on page 180.
Desc	Job description. For more information, see "DESC" on page 130.
Table	Scheduling library and table of the job. For more information, see "TABLE" on page 177.
Note	First line of the note (if one exists). For more information, see "NOTE" on page 187.

```
O Name Owner Odate Jobname JobID Typ ------ Status ------
T05SP115 T05 060601 / J0B Wait Schedule Group=NRD-GRP
CPU=A OrderID=0003E ===>BACKUP
SCHED-LIB=CTM.PROD.BKP(SPBKP)
*** Note ***
```

# **Display Type A (All Info)**

The following is an example of the All Info display type. In addition to the Default information, it contains statistical information about the job run.

Figure 41 Display Type A (All Fields)

```
Filter:
                 ----- CONTROL-M Active Environment ----- DOWN <A> - (3)
COMMAND ===>
                                                             SCROLL ==> CRSR
                  Odate Jobname JobID Typ ----- Status -----
O Name Owner
  DAILYPRD PRODMNGR 060601
                                          JOB Wait Schedule
   OrderID 001Q6 Grp CTM-CONTROL
                                Res. Use: Y
   Time Fr: Time Un:
                               Priority: 00
    Due-In: 0859 Due-Out: 0859 Late:
                       RBA: 000002
   Avg Elaps: 0000
  DAILYSYS SYSTEM 060601
                                        JOB Wait Schedule
   OrderID 001Q7 Grp CTM-CONTROL
                Time Un:
   MaxRC
                                Res. Use: Y
   Time Fr:
                               Priority: 00
    Due-In: 0859 Due-Out: 0859 Late:
  Avg Elaps: 0000 RBA: 000003
CTMLDNRS PRODMNGR 060601 JO
                                         JOB Wait Schedule
   OrderID 001Q8 Grp CTM-CONTROL
                Res. Use: Y
Time Un: Priority: 00
   Time Fr:
    Due-In: 0859 Due-Out: 0859 Late:
  Avg Elaps: 0000
CTMCLRES PRODMNGR 060601
                              RBA: 000004
                                         JOB Wait Schedule
Opt: ? Why L Log H Hold Z Zoom R Rerun A Activate O Force OK V View Sysout
     N Net D Del F Free S Stat G Group U Undelete J JCL Edit C Confirm 14.50.56
```

## Fields of the Display Type A (All Info)

Below is a description of fields in the All Info display type that do not appear in the Default display type. The fields that appear in the Default display type are described in the preceding section.

Table 56 Fields in the All Fields Display Type (part 1 of 2)

Field	Description
Grp	Name of the group.
MaxRC	Highest return code returned for the job.
Res. Use	Indicates (yes or no) whether the job uses (Control and/or Quantitative) resources.
Time Fr	Time the job began execution.
Time Un	Time the job ended execution.
Priority	Priority at which the job executed.
Due-In	Time by which the job was to be submitted.
Due-Out	Time by which the job execution must complete.

Table 56 Fields in the All Fields Display Type (part 2 of 2)

Field	Description
Late	Indicates if the job is late. Valid (Late) values:
	■ I – Late In - the job was submitted late
	<ul> <li>X - Late Execution - the job execution time was outside predefined limits</li> </ul>
	■ O – Late Out - the job ended outside the predefined time limit
Avg Elaps	Average elapsed run time of the job.
RBA	Relative Byte address of the job in the Active Jobs file.

## **Commands of the Active Environment Screen**

The commands in the following table can be requested by typing the command in the **COMMAND** field of the Active Environment screen and pressing **Enter**.

Table 57 Commands of the Active Environment Screen (part 1 of 7)

Command	Description
OPT	Toggles between the Primary Bottom line (composed of most of the available Primary commands) and the Alternate Bottom line (composed of all available options).
DISPLAY	Used to change display types. The format of the command is
	DISPLAY $x$ (abbreviated DI $x$ )
	where <i>x</i> is the desired display type. For example, DI A displays the All Info display type.
	<b>Note:</b> For a list of display types, enter <b>DISPLAY?</b> to show the Display Options window. To select a display type in the window, type S in the <b>Option</b> field next to the ID. To exit the window without selecting a display type, press the END key ( <b>PF03/PF15</b> ).
	Example
	DI N
	displays the <b>Net</b> fields display.
	Valid predefined displays are:
	<ul> <li>D – Default Display Type. For more information, see "Display Type D (Default)" on page 170.</li> <li>N – Net Fields Display Type. For more information, see "Display Type N (Network)" on page 243.</li> <li>A – All Fields Display Type. For more information, see "Display Type A (All Info)" on page 172.</li> </ul>

Table 57 Commands of the Active Environment Screen (part 2 of 7)

Command	Description
SHOW	Activates the specified screen filter, or opens the Show Screen Filter window or the Display Filters window, depending on the format of the command. For more information, see "Filtering the Active Environment Screen Display" on page 189.
	Valid formats are:
	<ul> <li>SHOW name - Activates the selected filter.</li> <li>SHOW? - Opens the Display Filters window that lists all available filters.</li> <li>SHOW (PF02/PF14) - Opens the Show Screen Filter window, displaying the settings of the currently active filter. This enables editing of the selected filter.</li> <li>SHOW name EDIT - Opens the Show Screen Filter window of the selected filter, displaying its settings. This enables editing of the selected filter.</li> </ul>
	<b>Note:</b> Reserved filter name DEFAULT can be used to activate or edit the default filter for the Active Environment screen.
	Only jobs conforming to selection criteria specified in the filter are displayed in the Active Environment screen
HISTORY	Displays the jobs in the History Jobs file in the History Environment screen, which is described in "History Environment Screen" on page 245.
RBAL	Scrolls the Active Environment screen so that the job with the specified relative byte address (RBA) is displayed at the top of the screen.
	The format of the command is
	RBAL <i>rba</i> (abbreviated RB <i>rba</i> )
	where <i>rba</i> is the relative byte address of the job.
	This command is especially useful for determining which job is using a particular resource. The RBA of the job using a resource is indicated in the IOA Conditions/Resources screen (Screen 4).

Table 57 Commands of the Active Environment Screen (part 3 of 7)

Command	Description
REFRESH	Initiates recalculation of job dependency information.
	The recalculation
	<ul> <li>updates the list of dependent jobs in the Job Dependency Network screen, recalculates logical dependencies for all job orders currently present in the Active Jobs file and updates the Job Dependency Network screen</li> <li>adjusts DUE OUT times for all job orders in the Active Jobs file that are not Held</li> <li>adjusts the priority of predecessor jobs</li> <li>simultaneously activates processes NET, DEADLINE, and PROPAGATE in the CONTROL-M monitor</li> </ul>
	<b>Note:</b> Although available in the Active Environment screen, this command is generally used in the Job Dependency Network screen.
AUTO	Activates AutoRefresh mode.
	The format of the command is
	AUTO n
	where <i>n</i> is any number of seconds within the range 1 through 99. For more information, see "AutoRefresh Mode" on page 100
JOBSTAT	Displays the Statistics screen, which provides statistics for the specified job. For more information, see "Statistics Screen" on page 238.
	Unlike Option S (STAT), which can be entered only for jobs appearing in the Active Environment screen, the JOBSTAT command can be entered for any job. Format of the command is
	JOBSTAT jobname groupname
	Specification of a group name is optional, but if no group name is specified, statistics are displayed only for jobs not belonging to any group.
SHPF	Displays the PFKey assignment of the screen. For more information, see "Commands and PFKeys" on page 93

Table 57 Commands of the Active Environment Screen (part 4 of 7)

Command	Description
NOTE	Displays or hides the first line of a note (if one exists).
	The format of the command is
	NOTE {ON OFF}
	where
	<ul> <li>ON – Displays the first line of the note</li> <li>OFF – Hides the note</li> </ul>
	If no ON of OFF qualifier is entered, the NOTE command toggles between displaying and hiding the first line of the note.
	The current setting is kept in the user profile for the next time the screen is displayed.
TABLE	Displays or hides the name of the job scheduling library and table from which the job was ordered.
	The format of the command is
	TABLE {ON OFF}
	where
	<ul> <li>ON – displays the name of the job scheduling library and table</li> <li>OFF – hides the name of the job scheduling library and table</li> </ul>
	If no ON of OFF qualifier is entered, the TABLE command toggles between displaying and hiding the name of the job scheduling library and table.
	The current setting is kept in the user profile for the next time the screen is displayed.

Table 57 Commands of the Active Environment Screen (part 5 of 7)

Command	Description
CPUID	Displays or hides the CPU ID of jobs subjected to dynamic resource acquisition. When displayed, the CPU ID on which the job is running (or executed on) appears after the job status.
	The format of the command is:
	CPUID {ON OFF}
	where:
	<ul> <li>ON – displays the CPU ID</li> <li>OFF – hides the CPU ID</li> </ul>
	If no ON or OFF qualifier is entered, the CPUID command toggles between displaying and hiding the CPU ID.
	The current setting is kept in the user profile for the next time the screen is displayed.
DESC	Displays or hides the description of each job.
	The format of the command is:
	DESC {ON OFF}
	where
	<ul> <li>ON – Displays the description</li> <li>OFF – Hides the description</li> </ul>
	If no ON or OFF qualifier is entered, the DESC command toggles between displaying and hiding the description.
	The current setting is kept in the user profile for the next time the screen is displayed.

Table 57 Commands of the Active Environment Screen (part 6 of 7)

Command	Description
DUMP	Used in special circumstances when requested by BMC Software Customer Support. The command is used to capture abends resulting from either internal or external events.
	The format of the command is:
	DUMP {ON   OFF}
	where
	<ul> <li>ON – provides a DUMP</li> <li>OFF – does not provide a DUMP</li> </ul>
	If no ON of OFF qualifier is entered, the DUMP command toggles between providing and not providing a DUMP.
	The current setting is kept in the user profile for the next time the screen is displayed.
	When DUMP ON is requested, the DUMP ON indicator is displayed in the first line of the screen.
GROUP	Displays or hides the group name. When displayed, the name of the group appears after the job status.
(PF11/PF23)	The format of the command is:
	GROUP {ON OFF}
	where:
	<ul> <li>ON – displays the group name</li> <li>OFF – hides the group name</li> </ul>
	If no ON or OFF qualifier is entered, the GROUP command toggles between displaying and hiding the group name.
	The current setting is kept in the user profile for the next time the screen is displayed.
OIDL	Scrolls the Active Environment screen so that the job with the specified order-ID is displayed at the top of the screen.
	The format of the command is:
	OIDL ord_ID
	where ord_ID is the order-ID of the job.

Table 57 Commands of the Active Environment Screen (part 7 of 7)

Command	Description
ORDERID	Displays or hides the order ID of each job.
	The format of the command is:
	ORDERID {ON OFF}
	where:
	<ul> <li>ON – Displays the order ID</li> <li>OFF – Hides the order ID</li> </ul>
	If no ON of OFF qualifier is entered, the ORDERID command toggles between displaying and hiding the order ID.
	The current setting is kept in the user profile for the next time the screen is displayed.
VIEW (PF04/PF16)	Displays the Global View screen, which provides a statistical overview of the status of jobs running under CONTROL-M. For more information, see "Global View Screen" on page 199.
VIEW GRAPH	Displays the View Graph screen, which provides a graphic statistical
	overview of the status of jobs running under CONTROL-M. For
(VG)	more information, see "View Graph Screen" on page 201.

# **Options of the Active Environment Screen**

Select an option by typing it in the **O** (**Option**) field to the left of the job order and pressing **Enter**. The following table describes the available options:

Table 58 Options of the Active Environment Screen (part 1 of 6)

Option	Description
? (Why)	Display the Why screen, which shows the reasons the job is in Wait Schedule status. For more information, see "Why screen" on page 205.
L (Log)	Display the Log screen, which shows all IOA Log messages for the specified job. For more information, see "IOA Log Screen" on page 296.
H (Hold)	Hold CONTROL-M operations on the job order. Only CONTROL-M operations concerning the job order are halted. The flow of the job through the operating system is not held. The HOLD request is recorded in the IOA Log file. The status of the job is changed to REQUESTED HELD. If the CONTROL-M monitor is active, the status changes to HELD. In some cases, a HOLD request may be rejected by the monitor.
Z (Zoom)	Display the Zoom screen, which "zooms in" on job details. For more information, see "Zoom Screen" on page 213.

Table 58 Options of the Active Environment Screen (part 2 of 6)

Option	Description
R (Rerun)	Rerun the job. A Rerun window is displayed. For more information, see "Confirm Rerun Window" on page 222.
A (Activate)	Reactivate a job or started task that has a status of either DISAPPEARED or FAILED REASON UNKNOWN. CONTROL-M searches the MVS/JES queues for the disappeared or failed job or started task.
	A job or started task is assigned a DISAPPEARED status if it has been accidentally deleted. Also, if JES is very busy, it sometimes sets the status of a job or started task to DISAPPEARED even though the job or started task actually exists.
	A job or started task is assigned a status of FAILED REASON UNKNOWN whenever CONTROL-M encounters a problem reading the SYSDATA files of the job and therefore cannot check the completion status of the job.
O (Force OK)	Force the job to complete with ENDED OK status. For more information, see "Force OK Confirmation Window" on page 247.
§Restart§ V (View Sysout)	View the execution history of the job in the Job Order Execution History screen. From this screen, the Sysout Viewing screen, which displays the archived SYSDATA of the job, can be requested. For more information on these screens, see "§Restart§Job Order Execution History Screen" on page 234, and "§Restart§ Sysout Viewing Screen" on page 236.
N (Net)	Display the Job Dependency Network screen, which shows all the predecessor and successor jobs for the selected job. For more information, see "Job Dependency Network Screen" on page 242.
D (Del)	Delete the job. For more information, see "Delete Confirmation Window" on page 211.
	<b>Note:</b> If you delete a Group entity, all jobs which are part of that Group are also deleted.
F (Free)	Free a held job order. All CONTROL-M operations for the job order are resumed. If the job is currently in the job queue of the operating system in HOLD state, the job is not released. The FREE request is recorded in the IOA Log file. The status of the job is changed to REQUESTED FREE. If the monitor is active, the FREE request is accepted after a few seconds.
S (Stat)	Display the Statistics screen, which shows job run statistics. Statistics for a job that is not in the Active environment can be displayed using command JOBSTAT. For more information, see "Statistics Screen" on page 238.

Table 58 Options of the Active Environment Screen (part 3 of 6)

Option	Description
G (Group)	Display the Group Entity (GRP entry) and all jobs that are part of that group. This option can be entered next to a GRP entry, or next to any job that is part of a group. Jobs that are part of a group are marked with the letter G to the right of the group name under display type A.  Option G must be entered as the last option in the screen.  When the Group option is requested, the name of the selected group appears in the title line of the screen.
U (Undelete)	Cancel a previously requested Delete. Valid only for jobs deleted by request. The job is returned to its status prior to the delete request.
	Note: If you undelete a job that is part of a deleted Group, the Group entity is undeleted, together with the individual job.  However, if you undelete a deleted Group, only the Group is undeleted, and not the jobs in the Group. If you want also to undelete a job or jobs that are part of that Group, you must undelete each job individually.
J (JCL Edit)	Edit the member that contains the JCL of the job.  By default, if the specified JCL member exists in the OVERLIB library, that member is edited. If the JCL member does not exist in the OVERLIB library, the member is edited in the MEMLIB library.
C (Confirm)	Confirm that this job is to be scheduled. A window is displayed to permit user confirmation. Entering Y sets the status of the job to WAIT SCHEDULE. For more information, see "Confirm Scheduling Window" on page 222.

Table 58 Options of the Active Environment Screen (part 4 of 6)

Option	Description
% (Simulation)	Simulates the action of the CONTROL-M Submission mechanism for a job that was previously placed in the Active Jobs file. This option is similar to the CTMAESIM utility, which is described in the section concerning testing AutoEdit syntax in Chapter 5, "JCL and AutoEdit Facility." The option produces the JCL stream for the job and a report of the process.
	The IOA Editor directs the output of the AutoEdit simulation to the user's screen, with the following header line displayed:
	CONTROL-M_AUTOEDIT_SIMULATION(memname)
	where <i>memname</i> is the name of the JCL member of the job.
	The report consists of two parts:
	<ul> <li>the messages produced by CONTROL-M during the simulated job processing</li> <li>the JCL stream as it would be submitted by the CONTROL-M Monitor to the MVS internal reader if the job is rerun</li> </ul>
	The user can use the IOA editor to edit the output, and save it as a member in a library.
	Notes:
	<ul> <li>For DUMMY jobs, no JCL stream is generated.</li> <li>To activate this function, the user must have read-access security authorization to the JCL library (MEMLIB).</li> </ul>

Table 58 Options of the Active Environment Screen (part 5 of 6)

Option	Description
B (Bypass)	Display the BYPASS option window. This option enables you to specify criteria and resources to be ignored for those jobs that have a status of WAIT SCHEDULE.
	By default, all fields in the BYPASS option window are set to N.
	You may set any or all these fields to Y, with the following effects:
	<ul> <li>Time Limit - All the time limit selection criteria of the job, such as TIME FROM, TIME UNTIL, DUE OUT, and NEXT, are ignored. The job is submitted when all other criteria are satisfied.</li> <li>IN Conditions - All IN conditions of the job are ignored. The job is submitted when all other criteria are satisfied.</li> <li>Quantitative Resources - All quantitative resources of the job are ignored. The job is submitted when all other criteria are satisfied.</li> <li>CONTROL Resources - All CONTROL resources of the job are ignored. The job is submitted when all other criteria are satisfied.</li> <li>Pipes - All PIPE statements of the job are ignored. The job is removed from the pipe sharing job collection of which it is part, and is submitted when all other criteria are satisfied.</li> <li>JCL - The member and library specified in the MEMNAME, MEMLIB, and OVERLIB statements of the job are ignored. When all run-time criteria are satisfied, a dummy job is submitted.</li> </ul>
	Note: When BYPASS JCL is specified, CONTROL-M handles post-processing of the job as if it were a dummy job and will ignore all ON PGMSTEP pgmstep DO blocks in the job.
	■ All BYPASS options – Enters Y in all the fields in the BYPASS option window.  When any BYPASS option has been set to Y, the status field of the job in the Active Environment Screen will show that the BYPASS feature is in use, with the relevant activated field identified. For example, the status may show BYPASS(Time + IN + QUANT), to indicate that the <b>Time Limit, IN Conditions</b> , and <b>Quantitative Resources</b> fields were set to Y, and that those criteria are being ignored.  If you set any field in the BYPASS window to Y, that setting only remains valid for the current run of the job. When the job is rerun or restarted, all <b>BYPASS</b> fields are reset to N.
	■ Using BYPASS does not require that the job be HELD. It is therefore possible that by the time CONTROL-M comes to handle the BYPASS request, the status of the job may no longer be WAIT SCHEDULE. If this occurs, the monitor ignores the BYPASS setting, and issues an appropriate message to the IOA log.
	<b>Warning:</b> You cannot perform BYPASS unless authorized to ZOOM and SAVE the job.

Table 58 Options of the Active Environment Screen (part 6 of 6)

Option	Description
W (MVBO/ Job Optimizer)	Display the MVBO/Job Optimizer screen for the selected job. Valid only for jobs under the control of MAINVIEW Batch Optimizer (MVBO). The MVBO/Job Optimizer screen is described in the MAINVIEW Batch Optimizer/Job Optimizer Reference Manual.
K (Kill)	(Only jobs that have Executing status) "Kills" the job, meaning that the job is cancelled (causing the job to fail with a system abend code of S222), and the status of the job is changed to ENDED NOTOK. Kill may not be specified for Started tasks or NJE jobs.
X (Exit)	Invoke user exit CTMX008. Placing an X next to a job in screen 3 keeps the details of the current job as input parameters for the exit.  For information about the function of user exit CTMX008, please consult with your site's IOA administrator.

### **Job Statuses**

The following job statuses can appear in the Active Environment screen:

Table 59 Job Statuses for the Active Environment screen (part 1 of 5)

Status	Description
ACTIVE	The job is a dummy job that has not yet reached the post-processing phase.
BUT NOT FOUND n TIMES	If a job is not found at least once, this status displays the number of times (n) CONTROL-M has looked for the job. If the job is still not found after 10 times, the status is changed to DISAPPEARED.
	Example
	JOB SUBMITTED BUT NOT FOUND 5 TIMES
	The job was submitted, but may have been purged. After checking 10 times, CONTROL-M changes the status to DISAPPEARED.
	<b>Note:</b> This default number can be changed by your INCONTROL administrator.
§Restart§ CLEANUP	Job is being run for Cleanup.
DELETED	The job order was deleted by an authorized user.
DISAPPEARED	Job disappeared completely. This status only occurs after a NOT FOUND status.
ENDED NOT "OK"	Job ended NOT OK.
ENDED NOT "OK" – ABENDED	Job abended.
ENDED NOT "OK" – DUE TO CC	Condition code that is not defined as OK has occurred.

Table 59 Job Statuses for the Active Environment screen (part 2 of 5)

Status	Description
ENDED NOT "OK" – FAILED – REASON UNKNOWN	This usually occurs following a system crash.
ENDED NOT "OK" – JCL ERROR	Job failed due to JCL error.
ENDED NOT "OK" – RERUN NEEDED	Rerun is needed for the job.
ENDED NOT "OK" – RERUN WAS NEEDED	Rerun was required for the previous execution of the job.
ENDED NOT "OK" – TERM ON NCT2	The job was terminated by CONTROL-M due to a NOT CATLGD 2 error.
ENDED "OK"	Job finished executing OK.
ENDED "OK" FORCED OK	Job ended OK due to a Force OK request.
EXECUTING	Job is executing.
EXECUTING (SYSOUT IN HOLD STATUS)	Job was placed in HOLD status by an operator issued JES HOLD command before CONTROL-M could read the job's output.
GOING TO START	Started task is eligible to be run and is about to be activated.
(GRP HELD)	The Group entity of which the job is part is in Held status, and as a result, the job itself is being logically held. (While the job's Group entity is being held, actions that require a Held status, such as Delete, Zoom, and Save, can be performed against the job. In addition, the CONTROL-M monitor does not handle the job. For example, if the job is in WAIT SCHEDULE status it is not selected for submission.)
HELD	Job is in hold status.
LATE	Job did not finish executing by the time specified in a SHOUT WHEN LATE statement.
LATE EXECUTION	The elapsed runtime of the job is outside the acceptable limits defined in a SHOUT WITHIN EXECTIME statement.
LATE SUBMISSION	Job was not submitted by the time specified in a SHOUT WHEN LATESUB statement.

Table 59 Job Statuses for the Active Environment screen (part 3 of 5)

Status	Description
NJE JOB	The job is not currently found in either Remote or Host node, but is in the process of transmission between nodes. (Either the job is being transmitted to the Remote node for execution, or the sysdata output of the job is being transmitted to the Host node.) CONTROL-M continues to search for the job until it is located on one of the nodes.
	BMC Software recommends that you do not purge jobs on the Remote node. However, if you do purge a job on the Remote node, you must notify CONTROL-M of the event by changing the value in the <b>NJE</b> field in the Active Environment Zoom screen (Screen 3.Z) to ' ' (Blank). After a short time, the job status changes to Disappeared.
NJE JOB (ID CHANGED)	The job ID of the NJE job has changed. When the job's sysdata output was transmitted back to the Host NJE node, the CONTROL-M monitor detected that the original job ID of the NJE job is occupied by another job. The CONTROL-M monitor continues to search for a job to match the new job ID.
NOT FOUND	Job not found in the queue. Check that the job or its sysout has not been accidentally deleted. This status may also appear when JES is very busy. In such a case, CONTROL-M waits for JES until it confirms that the job is lost.
NOT STARTED	Starting of the started task failed.
NOT SUBMITTED	Submission of the job failed.
NOTE	A Note has been added to the job, through the Zoom screen.
ON HST FILE	Job is currently in the History file. If the job is included in the flow of jobs being rerun, it will be restored to the Active Jobs file before being rerun.
	<b>Note:</b> This option appears only in the Rerun Flow Job List window.
ON OUTPUT QUEUE	Job is on the output queue of the remote NJE node or on the output queue of the host node with a changed job ID.
PRIOR RUN	Termination status of the previous job (or cyclic task) execution (for jobs that have been rerun).
PROBLEMS READING SYSOUT	Usually means that problems prevent the CONTROL-M monitor from reading the job's output.
PSEUDO	At the time when the job was ordered, CONTROL-M automatically converted it to a DUMMY job.

Table 59 Job Statuses for the Active Environment screen (part 4 of 5)

Status	Description
RELEASED	On Spool job has been released and is waiting to be executed.
REQUESTED CHANGE	Job parameters were changed using the Zoom option, but the request has not yet been performed by the CONTROL-M monitor.
REQUESTED FORCE OK	A Force OK request was issued for a held job, but the request has not yet been performed by the CONTROL-M monitor.
REQUESTED FREE	A free request was issued for a held job, but the request has not yet been performed by the CONTROL-M monitor.
REQUESTED HELD	A hold request was issued for the job, but the request has not yet been performed by the CONTROL-M monitor.
REQUESTED REACT	An activate request was issued for a job, but the request has not yet been performed by the CONTROL-M monitor.
REQUESTED RERUN	A rerun request was issued for the job, but the request has not yet been performed by the CONTROL-M monitor.
§Restart§ RESTARTED)	Job has run (executed) with the restart step under CONTROL-M/Restart (that is, a restart has been performed).
RESTORED	This job was restored from the History file. If this status is displayed in the History Environment screen, the job has been restored, rerun, and then copied to the History file as part of the New Day or a User Daily procedure.
RUN n	Run number. Incremented each time a cyclic task is executed or a job is rerun.
STARTED	Started task started, but is not yet in the operating system's job queue.
SUBMITTED	Job submitted, but is not yet in the operating system's job queue.
WAIT CONFIRMATION (FOR SCHEDULE)	Job is waiting for manual confirmation before it can be scheduled.
<b>§Restart§</b> WAIT CONFIRMATION (WITH RESTART)	Job is waiting for manual restart confirmation.
WAIT EXECUTION	Job is in the operating system's job queue waiting to be executed.
WAIT RELEASE	On Spool job is eligible to be run and is about to be released.
WAIT SCHEDULE	Job is waiting to be scheduled.
WAIT SCHEDULE ON SPOOL	Job is waiting to be scheduled but is already in input queue on spool.

Table 59 Job Statuses for the Active Environment screen (part 5 of 5)

Status	Description
WAIT SCHEDULE (PIPE)	For MAINVIEW Batch Optimizer (MVBO) users. Job is waiting to be scheduled and is a participant in a Pipe (Collection).
WAIT SUBMISSION	Job is eligible to be run and is about to be submitted.
§Restart§ (WITH RESTART)	The restart step under CONTROL-M/Restart will be added to the JCL of the job when the job is submitted (that is, a restart will be performed).

## **Group Statuses**

The following Group statuses can appear for the group entity in the Active Environment screen:

Table 60 Group Statuses for the Active Environment Screen

Status	Description
ACTIVE	All runtime criteria for the Group entity have been satisfied, but at least one job in the group has not ended and no job in the group has ended NOTOK.
ACTIVE - IN ERROR	All runtime criteria for the Group entity have been satisfied, but at least one job in the group has not ended and one or more jobs in the group ended NOTOK.
ENDED NOTOK	All jobs in the group have ended. At least one job ended NOTOK.
ENDED OK	All jobs in the group ended OK.
(ORDERING)	A Group entity has been ordered to the Active Jobs file, but not all of its jobs have been placed in the Active Jobs file, or connected to the Group entity. The ORDERING status disappears when all jobs in the Group appear in the Active Jobs file and are connected to the Group. Status ORDERING is an add-on to the Group's regular status.
REQUESTED DELETE	A delete request was issued for the Group entity, but the request has not yet been performed.

### Filtering the Active Environment Screen Display

Screen filters may be used to filter the Active Environment screen display.

A filter consists of a set of record selection criteria (selection fields and their values). Only records that conform to selection criteria specified in the filter are displayed on the screen.

The INCONTROL administrator may predefine filters and place them in the General profile.

Each user can activate an existing filter in the Active Environment screen by typing the command SHOW in the COMMAND line of the Active Environment screen.

The filtering feature utilizes two different windows, the Show Screen Filter window and the Display Filters window:

- Each user can define and name multiple filters for the screen by using the Show Screen Filter window, which is described in "Editing Filter Criteria" on page 192. User-defined filters are stored in the user profile. Filters that are kept in the user profile can be activated only by the user who defined the filter.
- Users can display the list of all available filters by opening the Display Filters window.

A predefined default filter (DEFAULT) is supplied for the Active Environment screen. Site-defined defaults determine whether the last filter used or the DEFAULT filter is activated upon reentry to the Active Environment screen.

### **Activating a Known Filter in the Active Environment screen**

The SHOW command may be used to activate an existing filter when you know the filter name. To activate an existing filter in the Active Environment screen, type the command SHOW in the COMMAND field, as follows:

SHOW name

where *name* is the name of the filter to be activated.

### **Displaying the List of Available Filters**

If you do not know the name of a filter, you can display the list of available filters in the Display Filters window. The display includes Global filters that are available to all users, and user-defined filters that are only available to the individual user.

You can then select a filter from the Display Filters window for activation or editing.

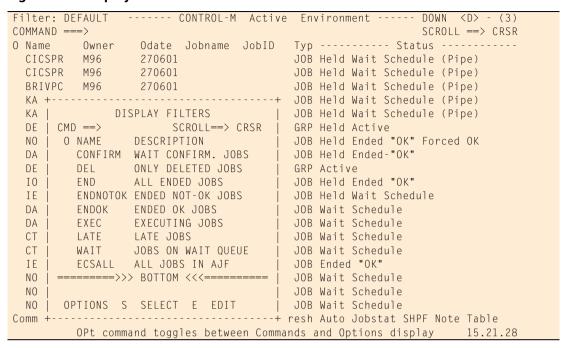
To open the Display Filters window, type the command **SHOW**? in the **COMMAND** field. The Display Filters window is opened.

#### NOTE -



Filters that are included with CONTROL-M are those appearing in the Display Filters window. These system-provided filters do not display descriptions.

Figure 42 Display Filters Window



The Display Filters window displays the following information:

Table 61 Field of the Display Filters Window

Field	Description
NAME	Name of the filter as it appears in the General or user profile.
DESCRIPTION	Description of the filter.

#### NOTE



When you create a user-defined filter and provide a description for that filter, the filter, and its description, are both displayed in the Display Filters window.

To select a filter in the list for activation or editing, type the appropriate option in the **O** (Option) field to the left of the filter name, and press **Enter**.

Table 62 Options of the Display Filters window

Option	Description
S (SELECT)	Activate the filter. The display of jobs in the Active Environment screen is filtered according to the filter criteria.
E (EDIT)	Display the filter's filtering criteria in the Show Screen Filter window to enable editing of the filter.

# **Editing Filter Criteria**

The Show Screen Filter window enables you to create or modify a filter.

- To open an existing filter for editing, either:
  - Type **SHOW** *name* **EDIT** in the Active Environment screen, where *name* is the name of the filter.
  - Type **E** (Edit) next to the filter name in the Display Filters window and press **Enter**.
- To edit the currently active filter, it is unnecessary to type the name or **Edit**. Just type **SHOW** in the **COMMAND** field and press **Enter**, or press **PF02/PF14**.
- To create a new filter, open any existing filter and enter a new name and description in the **FILTER** and **DESC** fields, which are described in Table 63 on page 193.

The Show Screen Filter window is displayed:

Figure 43 Show Screen Filter Window

```
------Show Screen Filter -----(3.SHOW)-
                   Save (Y/N) Desc:
Filter
Memname
Group
Wait Sched Y Wait time Y | Ended "OK" Y | Free Y Forced OK Y Wait Conf Y Wait Cond Y | Not "OK" Y | Held Y Grp Held Y Wait SUB Y Wait quant Y | Rerun Y | On Req Y CMEM Forc Y
Submitted Y Wait contrl Y | Disappeared Y | Deleted N Note Y Wait Exec Y Grp Active Y | Abended Y | Late N Restarted Y Executing Y | Unexpected CC Y | Pseudo N On Out Queue Y | JCL Error Y |
Task Type: Job Cyc Emr Stc Cst Est Ecj Ecs Wrn Grp
           Y Y Y Y Y Y Y
Resource Type: In Y Out Y Conds Y Resource Y Control Y
Owner
Odate: From
                             То
                                                Priority
Job
                             Appl
CPU Id
                 LPAR
Sch Lib
         OPt command toggles between Commands and Options display 11.07.49
```

### Fields of the Show Screen Filter Window

The Show Screen Filter window contains the following fields:

**Table 63** Fields of the Show Screen Filter Window

Field	Description
Filter	User-assigned name of the filter.
	The name entered in the <b>Filter</b> field may be modified.
	If there are unsaved changes to a filter in memory (discussed in "Closing the Show Screen Filter window" on page 198), an asterisk appears to the right of the filter name.
Save (Y/N)	Specifies whether to save modifications to the filter upon closing the window.
Desc:	User-defined description of the filter. The description entered here appears next to the name in the Display Filters window.

The fields listed in Table 64 define the selection criteria to be applied to the screen. Fill in these selection criteria as desired.

#### NOTE -



The selection criteria shown in Table 64 marked with the symbol ^P support masking. For more information, see "Character Masking" on page 83.

For compatibility with versions prior to version 6.2.xx, the SACTMSK profile variable can be used to specify how to treat character search criteria that do not use masking characters. The default setting (Y) results in CONTROL-M treating all character search criteria as if a masking character was entered. For example, if AB was entered as a member name to be searched, by default this entry is treated as AB*.

Table 64 Show Screen Filter Window Selection Criteria (part 1 of 5)

Criterion	Description
Memname ^P	Show only jobs of the specified member name. A maximum of five member names may be specified.
Group ^P	Show only jobs of the specified group. A maximum of two groups may be specified.

**Table 64** Show Screen Filter Window Selection Criteria (part 2 of 5)

Criterion	Description
status	Select only jobs that conform to the status selection criteria. Statuses are divided into three groups under the following headings: In Process, Ended, and State.
	The following logic applies to these headings:
	<ul> <li>Typing Y in the column heading enables further filtering of jobs on a status-by-status basis. Typing Y in a field in this column enables all jobs with this status to be shown, while typing N in a field in this column prevents jobs with this status from being shown.</li> <li>Typing N in the column heading filters out any job with a status listed under that heading, even if the status is marked Y or S.</li> </ul>
	The following logic applies to the fields under these headings:
	■ N/Y – If all status selection criteria are set to Y, all jobs and groups are displayed. Specifying a value of N for a specified status selection causes the groups and jobs for the specified status selection criteria not to be displayed. For example, if JCL Error is set to N, jobs that ended, or did not run, because of a JCL error are not displayed.
	Note that status selection criteria <b>Y</b> can be used to select jobs and groups on a very limited basis because of the interaction between different status selection criteria being set to <b>Y</b> . For example, in order to select all executing Late jobs, you must specify Executing=Y, FREE=Y, and Late=Y. However, this will cause all executing jobs to be displayed-even those that are not late.
	■ S – Select jobs to be displayed. Specifying a value of S for a specified status selection causes the groups and jobs for the specified status selection criteria to be displayed. Note that specifying a value of N for a specified status selection causes the groups and jobs for the specified status selection criteria not to be displayed. For example, to see only jobs and groups that are LATE and are executing, set LATE to S, set ENDED OK to N, set NOT OK to N, and set all other status selection criteria to Y.

**Table 64** Show Screen Filter Window Selection Criteria (part 3 of 5)

Criterion	Description				
status (continued)	<b>In Process</b> – This heading is for the status of jobs that are not yet finished.				
	Wait Sched	Jobs waiting to be scheduled			
	Wait Conf	Jobs waiting for confirmation			
	Wait Sub	Jobs waiting to be submitted			
	Submitted	Jobs submitted but not yet in queue			
	Wait Exec	Jobs waiting to be executed			
	Executing	Jobs that are currently executing			
	On Out Queue	Jobs on the output queue that have not yet been processed by CONTROL-M, for example, because of a system crash			
	Wait time	Jobs that were not submitted because the current time is outside the time limits specified in the job scheduling definition			
	Wait Cond	Jobs that are waiting for prerequisite conditions			
	Wait quant	Jobs that are waiting for Quantitative resources			
	Wait contrl	Jobs that are waiting for Control resources			
	Grp Active	Groups that are active, meaning <ul><li>■ all run time criteria of the group have been satisfied</li><li>■ at least one job in the group has not yet ended</li><li>■ no job in the group has ended NOTOK</li></ul>			
	Ended - This head	ding is for the status of finished jobs.			
	Ended "OK"	Jobs that ended OK			
	Not "OK"	Jobs that ended NOTOK			
	Rerun	Jobs that require rerun			
	Disappeared	Jobs that disappeared from the job queue			
	Abended	Jobs that abended			
	Unexpected CC	Jobs that ended with a condition code that is not defined as OK			
	JCL Error	Jobs that ended (or did not run) because of a JCL error			

**Table 64** Show Screen Filter Window Selection Criteria (part 4 of 5)

Criterion	Description			
status (continued)	State - This heading	ng is for the state of jobs and groups.		
	Free	Free jobs		
	Held	Held jobs		
	Deleted	Deleted jobs		
	On Request	Jobs for which a change in job status has been requested by a CONTROL-M user, but the request has not yet been processed by the monitor		
	Late	Jobs that were submitted, or finished executing, late; or jobs with an elapsed execution time outside specified limits		
	Pseudo	Jobs with prerequisite conditions that were adjusted as part of the "Adjust conditions in a group" feature		
	Forced OK	Jobs that ended OK due to a FORCE OK request		
	Grp Held	Groups that were held		
	CMEM Force	Jobs that were forced by the CMEM facility		
Task Type	Note	Jobs that contain a note that was added using the Zoom panel		
	Restarted	Jobs that were restarted under CONTROL-M/Restart		
	Limit the task types of jobs to be displayed. Valid task types are:			
	Job	Regular job		
	Сус	Cyclic job		
	Emr	Emergency job		
	Stc	Started task		
	Cst	Cyclic started task		
	Est	Emergency started task		
	Ecj	Emergency cyclic job		
	Ecs	Emergency cyclic started task		
	Wrn	Warnings. Supported for historical reasons		
	Grp	Group Entity		

**Table 64** Show Screen Filter Window Selection Criteria (part 5 of 5)

Criterion	Description	Description				
Res Name ^P	resource, Quar of two names r searched accor (described imn	An additional cross reference for all jobs that are using a Control resource, Quantitative resource, or prerequisite condition. A maximum of two names may be specified. The resources and conditions are searched according to those specified as Y (Yes) in Resource Type (described immediately below).				
Resource Type	Type of Resour of the Active E	rce or prerequisite condition to be used to filter the display nvironment screen.				
	In	All prerequisite conditions appearing in IN statements				
	Out	All prerequisite conditions appearing in OUT statements				
	Conds	All prerequisite conditions appearing in DO COND statements				
	Resource	All Quantitative resources				
	Control	All Control resources				
Owner ^P	Show only jobs be identified.	Show only jobs of the identified owner. A maximum of five owners may be identified.				
Odate	specified in the Date format is standard.	If a From date is specified without a To date, the current date is used as				
Priority	Show only jobs	Show only jobs with the specified priority.				
Pipe ^P	Show only job	participants in the specified pipe.				
Job ^P	Show only jobs	Show only jobs with the specified job name				
Appl ^P	Show only jobs value of APPL	Show only jobs with job scheduling definitions that contain the specified value of APPL				
CPU Id ^P	Show only jobs	Show only jobs that ran under the specified CPU				
LPAR ^P	Show only jobs	Show only jobs that are running or ran in the specified logical partition				
Sch Lib ^P	Show only jobs	Show only jobs with the specified schedule library				

# **Characteristics of the Show Screen filter**

The following rules govern the operation of the Show Screen Filter window:

■ The CONTROL-M monitor updates the status of jobs and groups displayed in the Active Jobs file (AJF). Therefore, if the CONTROL-M monitor is down, the status displayed in the AJF may not be up-to-date.

For example, if a job was waiting for a prerequisite input condition when the CONTROL-M monitor went down, the job will continue to be shown as waiting for the condition, even if the condition is present, until the CONTROL-M monitor updates the AJF.

- If you set Wait Sched to Y, the values set for Wait time, Wait Cond, Wait quant, and Wait contrl are ignored. CONTROL-M only looks at the values set for these fields if Wait Sched is set to N.
- Executing is only used for jobs (and not for groups).
- Grp Active is only used for groups (and not for jobs).
- If a job is part of a group, and the group is waiting for some criteria such as a time requirement, a prerequisite condition, and so on, the job has the same status as the group.

For example, if the group is waiting for a prerequisite condition, and the filter criteria is Wait Cond, the job will be displayed as waiting for the condition even if the condition is not required by the job scheduling definition of the job itself.

- If the job scheduling definition of a job requires it to wait for more than one input criteria, these criteria are not all checked at once. They are checked in the following sequence:
  - Wait time
  - Wait Cond
  - Wait quant
  - Wait contrl

For example, if the job is so defined that it must wait for a specific time and for a prerequisite condition, the job will be displayed in the list of jobs waiting for time, but will not be displayed in the list of jobs waiting for prerequisite conditions. However, when the time requirement is satisfied, if the prerequisite condition is not available, the job will be displayed in the list of jobs waiting for prerequisite conditions.

### **Closing the Show Screen Filter window**

The filter you have edited can be activated with or without saving changes, depending on the value entered in the **Save** field, as follows:

- To activate and save the filter, type Y (Yes) in the Save field. Changes to the filter are permanently saved.
- To activate the filter without saving it, type N (No) in the **Save** field. Changes are kept in memory only, but not saved.

After typing a value in the Save field, press one of the following keys:

**Table 65** Show Screen Filter Window - Closing Values

PFKey	Description
Enter	Filtering begins with the first job currently displayed in the screen and continues downward.
<b>PF07/PF19</b> (UP)	Filtering begins with the first job in the Active Job list and continues downward.
<b>PF08/PF20</b> (DOWN)	Filtering begins with the last job in the Active Job list and continues upward.

The window is closed and the filter is activated as defined or modified.

To cancel changes made in the Show Screen Filter window, use the RESET command (**PF04/PF16**). The changes are canceled regardless of the value entered in the **Save** field. The window is closed, and the filter that was previously in effect is restored.

By default, using the END command (PF03/PF15) in the window works like pressing Enter. However, the default may be modified so that END works like RESET.

# **Global View Screen**

The Global View screen is displayed by typing the command **VIEW** (abbreviated **V**) in the **COMMAND** field of the Active Environment screen and pressing **Enter**, or by pressing **PF10/PF22** in the Active Environment screen.

This screen provides a statistical overview of the status of the jobs running under CONTROL-M. Information is presented by GROUP name, by date (that is, separate statistics for the same group name on different dates).

#### NOTE -



All jobs having the same group name are grouped together, including jobs from different tables of different types.

Figure 44 Global View Screen

	GLOBAL	VIEW - E	BY GROU	JP	· · · · · · ( (	3.VIEW)
COMMAND ===>					SCROLL===	=> CRSR
TOTAL WAIT SCHEDULE	647 EXEC	JTING	19 E	END NOTOK	9 END OK	2014
STAT GROUP	ODATE #≀	WSC #EXC	#END	MEMNAME	JOB STATUS	S
WS CTM-CONTROL	060601	1	4	CTMCLRES	WAIT SCHEDULE	
ER PROD-ONSPOOL	060601		43	P0*	ENDED NOTOK	SOC4
* EN DD-DAY-PROD	060601		42			
WS BR-IVP-CC	060601	8	28	BRIVPCCE	WAIT SCHEDULE	
WS SYSTEMS-JOBS	060601	4	22	SMFCLEAN	WAIT SCHEDULE	
WS PROD-KPL	060601	47		PRDKPL01	WAIT SCHEDULE	
ER MT-PRODUCTION	060601	10	24	MTPRQV	ENDED NOTOK	SOC1
				MTRRU04	ENDED NOTOK	U0016
ER APPL-PROD-INTERNA	L 060601	9 2	2 2	INTPRD02	ENDED NOTOK	C0008
				INTPRD01	EXECUTING	
				INTPRD1A	WAIT EXECUTION	V
RN PR-PRODUCTION	060601	10	6 24		EXECUTING	
				PRDRPT99	EXECUTING	
				PRDDFN	EXECUTING	
				PRDRPT10	EXECUTING	
				PRDUPD12	EXECUTING	
				PRDUPD14	WAIT EXECUTION	V
RN VIJ-JOBS	060601	4	4 42		ENDED NOTOK	NOMEM
					ENDED NOTOK	
COMMANDS: REFRESH (	VIEW DATA)	EI	ND (RET	TURN TO ACT	TIVE SCREEN)	15.35.49

AutoRefresh mode is available under this screen.

To update the screen, press the REFRESH key (PF10/PF22).

To return to the Active Environment screen, press the END key (PF03/PF15).

# **Fields of the Global View Screen**

Table 66 Fields of the Global View Screen (part 1 of 2)

Field	Description
TOTAL	Displays the totals from the data. The following summary information is displayed for all jobs in the Active Jobs file except emergency jobs:  ■ WAIT SCHEDULE – Total number of jobs waiting to be scheduled ■ EXECUTING – Total number of jobs executing ■ END NOTOKG – Total number of jobs currently in ended NOTOK status ■ END OK – Total number of jobs that ended OK
Data lines	Display the following information about each group:

Table 66 Fields of the Global View Screen (part 2 of 2)

Field	Description			
STAT	Status of the group:			
	<ul> <li>WS - Wait Scheduling. All jobs are waiting to be scheduled (no jobs have begun running).</li> <li>ER - Error. At least one job has finished running and had an error.</li> <li>RN - Running. At least one job is running (executing) or has ended; not all jobs have finished executing; and no jobs have ended NOTOK.</li> <li>* EN - Ended OK. All jobs have finished running and ended OK.</li> </ul>			
GROUP	Name of the group.			
ODATE	Original scheduling date of the group, discussed in "ODATE" on page 153.			
#WSC	Number of jobs in Wait Schedule state.			
#EXC	Number of jobs executing (or in the input queue).			
#END	Number of jobs that have finished executing.			
MEMNAME	Name of each active member (job) in the group. The members that are displayed are those			
	<ul><li>executing (or in the input queue)</li><li>ended NOTOK</li></ul>			
	If none of the above is found within the group, the first job that is waiting to be scheduled is displayed.			
JOB STATUS	Status of each job in the group. In case of error, the type of error is shown (for example, abend code).			

# **View Graph Screen**

The View Graph screen is displayed by typing the VIEW GRAPH command (abbreviated  $V\ G$ ) in the **COMMAND** field of the Active Environment screen and pressing **Enter**.

This screen provides a statistical overview of the status of the jobs running under CONTROL-M, in graph form. Information is presented by GROUP name.

#### - NOTE -



All jobs having the same group name are grouped together, including jobs from different tables of different types.

AutoRefresh mode is available under this screen.

To update the screen, type the command **REFRESH** and press **Enter**, or press **PF04/PF16**.

To return to the Active Environment screen, press the END key (PF03/PF15).

Two formats for the View Graph screen are available, one for color displays and one for non-color displays. They are discussed on the following pages.

# **View Graph Screen Format for Color Terminals**

Figure 45 View Graph Screen Format for Color Terminals

		VIEW GRAPH - BY GROUP(3.GRAPH)
COMMAND ===>		SCROLL==> CRSR
TOTAL WAIT SCHEDULE	674	EXECUTING 28 END NOTOK 11 END OK 1549
		%+80+-100%
EBD-PRODUCTION	27	B100BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
		G100GGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGG
MT-PRODUCTION	40	B100BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
VEJ-JOBS	39	G100GGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGG
PROD-KPL	16	B100BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
INTER-PRODUCTION	42	RR50RRRRRRRRRRRRRRRRRRRRRGG50GGGGGGGGGG
NTN-APPLICATION	35	B100BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
APPL-PROD-INTERNAL	37	B100BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
CLIENTS-STATEMENTS	38	R100RRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRR
PR-PRODUCTION	40	B100BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
BR-IVP-CC	10	B100BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
SYSTEMS-JOBS	36	B100BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
CICS-BATCH-JOBS	28	G100GGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGG
DD-NIGHT-PROD	37	B100BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
BKP-PROD-L	10	BB20BBBBBYY20YYYYYRR40RRRRRRRRRRRRRRRRRGG20GGGGGG
BKP-PROD-ACCOUNT	-	B100BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
BKP-PROD-DD	14	R100RRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRR
FDS-JOBS	39	R100RRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRR
JJL-NIGHT-PROD	33	B100BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
COMMANDS: REFRESH	(VIEW [	DATA) END (RETURN TO ACTIVE SCREEN) 01.26.56

# **Fields of the View Graph Screen**

Table 67 Fields of the View Graph Screen (part 1 of 2)

Field	Description
TOTAL	Displays the totals from the data. The following summary information is displayed for all jobs in the Active Jobs file except emergency jobs:
	<ul> <li>WAIT SCHEDULE – Total number of jobs waiting to be scheduled</li> <li>EXECUTING – Total number of jobs executing</li> <li>END NOTOKG – Total number of jobs currently in ended NOTOK status</li> <li>END OK – Total number of jobs that ended OK</li> </ul>
The data lines di	splay the following information for each group:

Table 67 Fields of the View Graph Screen (part 2 of 2)

Field	Description
GROUP NAME	Name of the group.
SUM	Total number of jobs in the group.
JOB GRAPH	Job graph indicates the number of jobs in each status, in each group.
Scale	Scale line used to simplify reading the percentage of jobs of each status in the group. The scale used (that is, the number of jobs represented by each column) automatically adjusts based on the number of jobs in the group containing the most jobs.

# **Job Graph**

In the job graph (D), job statuses are differentiated by color, as follows:

#### **NOTE**



Because this guide is printed in black and white, the different colors in the screen are represented by different shadings in this guide.

**Table 68 Job Status Color** 

Color	Description
Blue	WAIT SCHEDULE
Yellow	EXECUTING
Red	END NOTOK
Green	END OK

For each group in the graph, the number of columns of a particular color depends on the number of jobs having that status.

# **View Graph Screen Format for Non-Color Terminals**

Figure 46 View Graph Screen Format for Non-Color Terminals

		VIEW GRAPH - BY GROUP(3.GRAPH)
COMMAND ===>		SCROLL==> CRSR
TOTAL WAIT SCHEDULE	674	EXECUTING 28 END NOTOK 11 END OK 1549
		+20+40+60+80+-100%
FBD-PRODUCTION		\$100\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$
GPI - PRODUCTION		% 1 0 0 % % % % % % % % % % % % % % % %
MT-PRODUCTION		\$100\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$
VFJ-JOBS	39	% 1 0 0 % % % % % % % % % % % % % % % %
PROD-KPI		\$100\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$
INTER-PRODUCTION	42	**50*********************************
NTN-APPLICATION	35	\$100\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$
APPL-PROD-INTERNAL	37	\$100\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$
CLIENTS-STATEMENTS	38	*100********
PR-PRODUCTION	40	\$100\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$
BR-IVP-CC	10	\$100\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$
SYSTEMS-JOBS	36	\$100\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$
CICS-BATCH-JOBS	28	%100%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
DD-NIGHT-PROD	37	\$100\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$
BKP-PROD-L	10	\$\$20\$\$\$\$\$++20+++++**40***************%20%%%%%%
BKP-PROD-ACCOUNT	9	\$100\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$
BKP-PROD-DD	14	*100********
FDS-JOBS	39	*100*********
JJL-NIGHT-PROD	33	\$100\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$
COMMANDS: REFRESH	(VIEW D	DATA) END (RETURN TO ACTIVE SCREEN) 01.26.56

# **Fields of the View Graph Screen**

Table 69 Fields of the View Graph Screen

Field	Description
TOTAL	Displays the totals from the data. The following summary information is displayed for all jobs in the Active Jobs file except emergency jobs:
	<ul> <li>WAIT SCHEDULE – Total number of jobs waiting to be scheduled.</li> <li>EXECUTING – Total number of jobs executing.</li> <li>END NOTOKG – Total number of jobs currently in ended NOTOK status.</li> <li>END OK – Total number of jobs that ended OK.</li> </ul>
GROUP NAME	Name of the group.
SUM	Total number of jobs in the group.
JOB GRAPH	Job graph indicates the number of jobs in each status, in each group.
Scale	Scale line used to simplify reading the percentage of jobs of each status in the group. The scale used (that is, the number of jobs represented by each column) automatically adjusts based on the number of jobs in the group containing the most jobs.

The data lines display the following information about each group:

#### **Job Graph**

In the job graph, job statuses are differentiated by symbols, as follows:

**Table 70 Job Graph Status Symbols** 

Symbol	Description
\$	WAIT SCHEDULE
+	EXECUTING
*	END NOTOK
%	END OK

For each group in the graph, the number of columns containing a particular symbol depends on the number of jobs having that status.

# Why screen

The Why screen (Screen 3.?) is displayed when the ? (Why) option is entered on the Active Environment screen. The Why screen shows the reasons why a job is in WAIT SCHEDULE status.

Figure 47 Active Environment Why Screen

```
PRUPDTO2 SCHEDULING ANALYSIS -----(3.?)
COMMAND ===>
                                                            SCROLL===> CRSR
OPT DESCRIPTION
                   FROM 1730
    TIME LIMIT
                                         UNTIL 0200
                   DB2-POWER
   RESOURCE
                                         QUANTITY 0030
                             HELD BY PRDKPL01 (U) QUANTITY 0022
                             HELD BY GPLIR17A (U) QUANTITY 0020
                            HELD BY INTROOOZ (U) QUANTITY 0010
HFLD BY PRUPDOLV (U) QUANTITY 0010
OHANTITY 0025
                   CARTRIDGE
    RESOURCE
                                          QUANTITY 0002
                             HELD BY
                                      PRDKPL01 (U) QUANTITY 0001
                             HELD BY GPLIR17A (U) QUANTITY 0002
   IN HOLD STATE
   CONDITION PRUPDTO1-ENDED-OK
                                                    ODATE 0606
   NOT-COND PRPLD03-ENDED-NOTOK
                                                    ODATE 0606
   GROUP SCHEDULING ANALYSIS FOR GROUP ACCOUNT (ACCOUNT-GROUP)
   GROUP'S RUNTIME CRITERIA SATISFIED
                            END OF "WHY" LIST <<<<<<<<<<<<<<<<<<<<<<<
 OPTION: A ADD CONDITION D DELETE NOT-COND
                                                                  10.32.27
```

To return to the Active Environment screen, press the END key (PF03/PF15).

#### Possible WHY reasons are:

- ALL RUNTIME CRITERIA SATISFIED. JOB WILL BE SUBMITTED SOON
- CONTROL-M MONITOR IS NOT ACTIVE
- IN HOLD STATE
- WAIT CONFIRMATION
- TIME LIMIT FROM hhmm UNTIL hhmm
- NEXT RUN FROM mmddyy hhmm
- CONDITION condition-name ODATE mmdd

Prerequisite condition required by the job, along with its original scheduling date.

■ RESOURCE resource-name [R] QUANTITY quantity BY priority memname

Name and quantity of a Quantitative resource not currently available for the job. For critical path jobs, a job with a higher path priority than the current job is also identified.

■ CONTROL OVER resource TYPE type BY priority memname [ownership type]

Name and type of a Control resource currently being used by another job order, which is identified by name. For critical path jobs, path priority of the owner is also identified.

■ CONTROL OVER resource TYPE type HELD BY priority *********** [ownership type] IOAID ioaid

Name and type of a Control resource currently being used by another job order in a different instance of the CONTROL-M monitor. The IOAID of the monitor holding this resource is also shown.

JOB WAIT FOR PIPES COLLECTION

PIPE pipename

The job was not run for one of the following reasons:

- CONTROL-M is waiting for the minimum number of participants in the indicated pipe.
- At least one prerequisite (prerequisite condition, resource, confirmation, and so on) for a participant in the indicated pipe is not satisfied.

If the job belongs to a Group scheduling table, the Why screen displays messages related to both the selected job and the group to which the job belongs. In this case, the reasons indicated above may be applicable to the selected job and/or to the group.

To enable you to distinguish between "job" reasons and "group" reasons, the job reasons appear in the screen before the group reasons, and the two sets of reasons are separated by the following line:

GROUP SCHEDULING ANALYSIS FOR GROUP group-memname (groupname)

In addition to the above line, the following reasons can appear only for a job in a Group scheduling table:

■ JOB'S RUNTIME CRITERIA SATISFIED

This reason applies to the job.

■ GROUP'S RUNTIME CRITERIA SATISFIED

This reason applies to the group.

### Condition display modes in the Why screen

If a job is waiting for prerequisite conditions, the conditions can be displayed in the Why screen in two different modes:

- In the Missing Conditions display mode, only the conditions which the job is actually waiting for are displayed. Logical operators and parentheses are not displayed.
- In the All Conditions display mode, all the prerequisite conditions defined for the job are displayed, along with all the relevant logical operators and parentheses.

Conditions for which the job is not waiting because they already exist are displayed with the string 'SATISFIED' following their original scheduling date. Conditions for which the job is not waiting, even though they do not exist, are displayed with the string 'REDUNDANT' following their original scheduling date. To switch between the display modes, use the CNDSHOW command. Typing CNDSHOW MISSING switches to the Missing Conditions display mode, and typing CNDSHOW ALL switches to the All Conditions display mode.

The condition display mode upon entry to the screen is determined by the value of the SWHYCND variable in the user's profile: If the value of SWHYCND is M, then the Missing Conditions display mode is used (this is also the default, if the SWHYCND variable is not found the user's profile). If the value of SWHYCND is A, then the All Conditions display mode is used.

## Adding conditions in the Why screen

If the Why screen indicates that a job is waiting for prerequisite conditions, the indicated conditions can be manually added using the Why screen by typing **A** (Add Condition) in the **OPT** (Option) field next to the condition.

Specify option **A** for every condition to be added, and press **Enter**.

When adding conditions, a confirmation window may be displayed depending on user profile customization. The confirmation window is described in "The Why Screen Add Condition or Delete NOT-COND Confirmation Window" on page 208.

# **Deleting negative conditions in the Why screen**

A negative or inverted condition is a condition that prevents a job from running.

Negative conditions can be seen in the Why screen. They can be identified by the description NOT-COND.

If the Why screen indicates that a job is waiting for a NOT-COND (negative condition) that is preventing the job from running, the NOT-COND can be deleted manually. This enables the job to run despite the fact that the NOT-COND is true.

To delete a NOT-COND manually, type  ${\bf D}$  (Delete NOT-COND) in the  ${\bf OPT}$  (Option) field next to the condition.

Type **D** for every NOT-COND to be deleted, and press **Enter**.

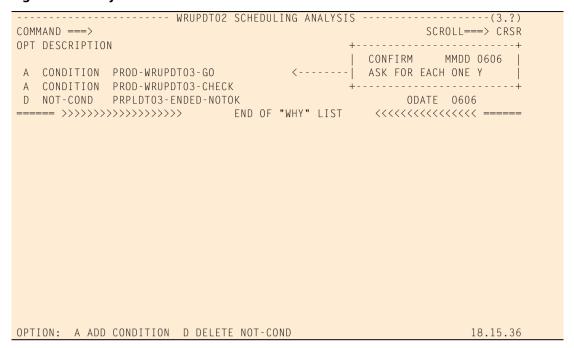
When deleting NOT-COND conditions, a confirmation window may be displayed depending on user profile customization. The confirmation window is described in the following paragraphs.

# The Why Screen Add Condition or Delete NOT-COND Confirmation Window

When adding conditions or deleting NOT-COND conditions, a confirmation window may be displayed depending on user profile customization:

- By default, when **A** or **D** is entered in the Why screen, a confirmation window is displayed only when the date reference of the condition is **** or \$\$\$\$. Addition or deletion of conditions without generic date references is performed without confirmation from the user.
- If, however, the user profile has been customized accordingly, the following confirmation window is always displayed when either **A** or **D** is entered.

Figure 48 Why Screen Add Condition or Delete NOT-COND Confirmation Window



Fill in or modify the fields of the confirmation window as follows and press Enter.

Table 71 Fields of the Add Condition or Delete NOT-COND Confirmation Window (part 1 of 2)

Field	Description
CONFIRM	Confirms whether to process the Add Condition or Delete NOT-COND request. Valid values are:
	<ul> <li>Y (Yes) – Process the Add Condition or Delete NOT-COND request.</li> </ul>
	■ N (No) – Cancel the request.
date	Date of the listed condition.
	■ If the date reference of the listed condition contains **** or \$\$\$\$, the date field of the window is unprotected and you must explicitly enter the date in the date field.
	■ If the date reference of the listed condition is a specific date (in either mmdd or ddmm format, depending on the standard in use at the site), the date field of the window is protected and its value cannot be changed.

Table 71 Fields of the Add Condition or Delete NOT-COND Confirmation Window (part 2 of 2)

Field	Description
ASK FOR EACH ONE	This line is displayed only if more than one Add Condition or Delete NOT-COND is requested. It determines whether individual confirmation is required for each Add Condition or Delete NOT-COND request. Valid values are:  Y (Yes) – Individual confirmation is required for each Add Condition or Delete NOT-COND request. The specified CONFIRM value (Y or N) applies only to the current Add Condition or Delete NOT-COND request.
	■ N (No) – Individual confirmation is not required for each Add Condition or Delete NOT-COND request. The specified CONFIRM operation is applied to all Add Condition and Delete NOT-COND requests. (If CONFIRM is Y, all Add Condition and Delete NOT-COND requests are processed; if CONFIRM is N, no Add Condition or Delete NOT-COND requests are processed.)

# **Deleting a Job**

CONTROL-M only allows deletion of WAIT SCHEDULE jobs, or jobs that have finished executing. To be deleted, a job must be in HELD status. The deletion request is recorded in the IOA Log file. The job is logically deleted, that is, flagged as deleted, from the Active Jobs file immediately. It is physically deleted from the disk the next time cleanup (for example, New Day processing) is performed.

#### NOTE



Logically deleted jobs can be undeleted by option U (Undelete). When jobs are undeleted they are added back to the Active Jobs file with the same status they had prior to deletion.

To delete a job, type **D** (Delete) in the **Option** field to the left of the job and press **Enter**.

# **Deleting Critical Path Jobs**

Critical path jobs (even in HELD status) that hold a Control or Quantitative resource can only be deleted through the following steps:

- 1 Remove the critical path priority of the job using the Zoom screen (Screen 3.Z).
- 2 Free the job.

- 3 When the job reverts to WAIT SCHEDULE status, hold the job.
- **4** Delete the job.

### **Deleting Group Entities**

When a Group entity is deleted, all the jobs belonging to that group are deleted. When a group is deleted, and a job within that group is undeleted, the Group entity itself is undeleted together with the job. To delete a group of jobs, the Group entity containing the jobs must first be in WAIT or END status. Place the Group entity in HELD status. Once a Group entity is in HELD status, you can delete the Group entity.

If all the jobs within a group are deleted, you can delete the Group Entity itself through the following steps:

- 1 Put the group on hold. If it is already held, free it, then put it on hold again.
- **2** Delete the Group Entity.

The reason for freeing a held group and then putting it on hold again before attempting to delete it is that, for efficiency, when a group is held, CONTROL-M does not check the status of the jobs in it.

#### **Delete Confirmation Window**

When requesting job deletions, a Delete Confirmation window may be displayed, depending on user profile customization:

- By default, when option D is entered in the Active Environment screen, deletion requests are performed without confirmation from the user.
- If, however, the user profile has been customized accordingly, the following Delete Confirmation window is displayed, in sequence, for each deletion request.

Figure 49 Active Environment Screen Delete Confirmation Window

```
Filter:
               ----- CONTROL-M Active Environment ----- UP
COMMAND ===>
                                                     SCROLL ==> CRSR
                Odate Jobname JobID
                                    Tvp ------ Status -----
O Name
         Owner
        PROD 060601
                                    JOB Wait Schedule (Pipe)
 IEFBR14T TEST 060601 M70TEST0/24897
                                    JOB Ended "OK"
 PRD1 PROD 060601
                                   JOB Wait Schedule (Pipe)
                +----+ Ended "OK"
 IEFBR14T
D SELIGRP <-----| Delete (Y/N) | Ended- Not "OK"
                +----+ Ended "OK"
 GRPJ0B1
 GRPJOB2 TEST 060601 M70TEST2/24929 JOB Ended "OK" GRPJOB3 TEST 060601 M70TEST3/24930 JOB Ended Not "OK" - Abended
 Opt: ? Why L Log H Hold Z Zoom R Rerun A Activate O Force OK V View Sysout
    N Net D Del F Free S Stat G Group U Undelete J JCL Edit C Confirm 16.28.18
```

Fill in the window as follows and press Enter.

To confirm the delete request, type **Y** (Yes) in the window.

To cancel the delete request, type N (No) in the window.

# Log Screen

To display the Log screen, type option **L** (Log) in the Active Environment screen. The Log screen displays all Log messages of the specified job.

Figure 50 Active Messages Log Screen

```
FILTER:
                    -- LOG MESSAGES FOR JOB(S) INTROOO4 -----(3.LOG)
COMMAND ===>
                                                                 SCROLL ===> CRSR
SHOW LIMIT ON ==>
                                                            DATE 060601 - 060601
     TIME ODATE USERID CODE ----- M E S S A G E -------
060601 131143 060601 M22
                              JOB511I JOB INTROOO4 ODATE 060601 ID=00019 PLACED
                                       ON ACTIVE JOBS FILE
060601 131148 060601 M22
060601 131150 060601 M22
060601 131651 060601 M22
                              SEL203I JOB INTROOO4 ELIGIBLE FOR RUN
                              SUB133I JOB INTROOO4 SUBMITTED
                              SPY281I JOB INTROOO4 INTROOO4/04371 START
                                       98253.1316 STOP 98253.1316 CPU OMIN
                                       00.04SEC SRB OMIN 00.00SEC 0.19 9QFDSF
060601 131651 060601 M22
                               SPY254I JOB INTROOO4 INTROOO4/04371 SCANNED
060601 131652 060601 M22
                              SEL206W JOB INTROOO4 INTROOO4/04371 ABENDED CC
                                       SB37 STEP STEP01
060601 131652 060601 M22
                               SEL219I JOB INTROOO4 INTROOO4/04371 ENDED "NOT
                                       0K"
060601 132814 060601 M22
                              CTM659I RERUN OF TASK INTROOO4 ODATE 060601
                                       PERFORMED
060601 132817 060601 M22 SEL220I JOB INTROUU4 WILL BE REMARKS
SEL203I JOB INTROOO4 ELIGIBLE FOR RUN
SEL203I JOB INTROOO4 SURMITTED
060601 132818 060601 M22
                              SUB133I JOB INTROOO4 SUBMITTED
NO MORE LOG MESSAGES
                                                      CMDS: SHOW, GROUP, CATEGORY, SHPF
                                                                        13.24.01
```

Usage of the Log screen is explained in detail in "IOA Log Screen" on page 296. However, if you entered the Log screen by option L on the Active Environment screen instead of by option 5 on the IOA Primary Option menu, note the following differences in usage:

- The SHOW command cannot be used with any parameters or qualifiers.
- Only filter options related to CONTROL-M (and CMEM) are displayed in the Show Screen Filter window.
- Only the default job filter can be displayed.

If you enter **L** (Log) in the O (Option) column for multiple jobs in the Active Environment screen, the log displays are stacked. Each time the END key (**PF03/PF15**) is pressed, the next log in the stack is displayed, until all logs have been displayed.

To return to the Active Environment screen, press **END** (**PF03/PF15**).

### **Zoom Screen**

The Zoom screen "zooms in" on the details of a specific job order. To display the Zoom screen, type **Z** (Zoom) on the Active Environment screen.

#### — NOTE -



To save changes made in the Zoom screen, the job must be placed in HELD state before entering the Zoom screen.

Figure 51 CONTROL-M Zoom Screen

(3.Z)
COMMAND ===> SCROLL==> CRSR
MEMNAME PRDKPLO1 MEMLIB CTM.PROD.JOBLIB  OWNER M44 TASKTYPE JOB PREVENT-NCT2 DFLT N  SCHDTAB MIKLE SCHDLIB CTMP.V524.SCHEDULE  APPL PROD GROUP KPL  OVERLIB STAT CAL PERIOD  SCHENV SYSTEM ID NJE NODE  JOBNAME JOBID ODATE 060601 ORDERID 0005C MAXWAIT 04  RESTART DECISION-FROM . TO . CONFIRM N  DESC DAILY PRODUCTION - START OF PRODUCTION GROUP KPL  SET VAR  CTB STEP AT NAME TYPE  DOCMEM PRDKPL01 DOCLIB CTM.PROD.DOC
NOTE
DOC
IN DAILY-PROD-KPL-GO 0606  CONTROL DB2-MAIN-FILE E  RESOURCE INIT 0001 CART 0001  PIPE CTM.PROD.PIPE  FROM TIME + DAYS UNTIL TIME + DAYS PRTY CONFIRM  DUE IN + DAYS ELAPSE DUE OUT + DAYS  TIME ZONE: WAIT FOR ODATE:  CPU-ID NODE NAME NJE SEARCH COUNTER LPAR 0S35
OUT PROD-PRDKPLO1-OK 0606 + AUTO-ARCHIVE Y SYSDB Y MAXDAYS MAXRUNS RETENTION: # OF DAYS TO KEEP # OF GENERATIONS TO KEEP SYSOUT OP C (C,D,F,N,R) 2 FROM MAXRERUN MEMBER INTRVL FROM END NXT RUN STEP RANGE FR (PGM.PROC) . TO . ON PGMST PROCST CODES A/O * DO SHOUT WHEN NOTOK TIME + DAYS TO OPER2 URGN R MS DAILY PRODUCTION JOB PRDKPLO1 ENDED NOT OK. NOTIFY PRODUCTION MANAGER SHOUT WHEN TIME + DAYS TO URGN MS
APPL TYPE  APPL FORM  INLINE JCL: N
>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

#### – NOTE



The Zoom screen is displayed in Browse mode if the requested job order is already being zoomed by another user. In this case, updates are not permitted.

The Zoom screen is similar to the Job Scheduling Definition screen used for defining job production parameters, but it is different in several respects:

- The Zoom screen contains fields that do not appear in the Job Scheduling Definition screen (and vice versa).
- Some fields on the Zoom screen cannot be modified at all. Other fields can or cannot be modified depending on the status of the job.
- Changes to a field in the Zoom screen affect only the current job order, not the job scheduling definition.

For information about most fields in the Zoom screen, see "Job Scheduling Definition Screen – Defining Schedules" on page 127.

Fields of the Zoom screen that are not in the Job Scheduling Definition screen are described below:

Table 72 Fields of the Job Scheduling Definition Zoom Screen (part 1 of 4)

Field	Description
SCHDTAB	Name of the scheduling table from which the job was ordered.
SCHDLIB	Name of the scheduling library from which the job was ordered.
ORIGLIB	The original value of the MEMLIB parameter before CONTROL-M changed it to DUMMY.
	This line appears only in PSEUDO jobs, meaning jobs in a group that were automatically changed by CONTROL-M into DUMMY jobs when they were ordered.
	For more information on PSEUDO jobs, see "ADJUST CONDITIONS: General Job Parameter" on page 389.
STAT CAL PERIOD	1-character field that identifies the actual days within the CONTROL-M periodic calendar that were used in calculating statistics relating to the job.
JOBNAME	Name of the job (available only after job submission).
JOBID	Job number (available only after job submission).
ODATE	Original scheduling date assigned to the job.
ORDERID	Unique job order ID in CONTROL-M.

Table 72 Fields of the Job Scheduling Definition Zoom Screen (part 2 of 4)

Field	Description
§Restart§ RESTART DECISION	This has the following subparameters:
DECISION	■ FROM – Program step (and, optionally, procedure step) names at which to begin processing the job restart.
	■ TO – Program step (and, optionally, procedure step) names at which the restarted job terminates processing. This parameter is optional. If the FROM parameter is specified and the TO parameter is not specified, the job is rerun until the last step.
	■ CONFIRM – Valid values are:
	<ul> <li>Y (Yes) - If the job is to be resubmitted as a result of a DO RERUN statement, manual confirmation is required (using the Active Environment screen).</li> <li>N (No) - If the job is resubmitted as a result of a DO RERUN statement, manual confirmation is not required.</li> </ul>
NOTE	Text of a note has been added to the job order. For more information, see "Adding or Editing a Job Order Note" on page 220.
IN	For users who have MVBO (Mainview Batch Optimizer) installed.
	If an IN condition name is preceded by the non-modifiable string IGN, that IN condition will be ignored when evaluating the job's run time criteria. The condition is ignored to enable an entire collection of PIPE-sharing jobs to run simultaneously.
DUE IN + DAYS	For complete details, see "DUE OUT: Runtime Scheduling Parameter" on page 492.
ELAPSE	Anticipated elapse time (that is, anticipated job execution time). The value used is the average of the run times of the job in the CONTROL-M Statistics file.
WAIT FOR ODATE	Whether a job can be executed even though ODATE is a future date. Valid values are:
	<ul> <li>Y - The job cannot be executed until ODATE arrives.</li> <li>N - The job can be executed even if ODATE has not yet arrived.</li> </ul>
	When the Job Scheduling Definition Zoom screen is displayed, the value that appears in this field varies as follows:
	<ul> <li>When the CTMJOB utility was used to order the job, if the ODATEOPT parameter was set to RUN, the value is Y.</li> <li>If the job was pre-ordered using the Time Zone feature in the New Day procedure, and the ODATEOPT parameter was automatically set to RUN, the value is Y.</li> <li>If the job was ordered or forced from the Job List Screen, and the WAIT FOR ODATE field in the Job List Exit Option window was set to Y, the value that appears in the Zoom screen is also Y.</li> </ul>
	You can change the value that appears in this field.

Table 72 Fields of the Job Scheduling Definition Zoom Screen (part 3 of 4)

Field	Description
CPU-ID	CPUID on which the job executes (if \$ Quantitative resources were specified). This field contains the selected \$ value, that is, the CPUID. For more details, see "RESOURCE: Runtime Scheduling Parameter" on page 596.
NODE NAME	Node on which the job executes (as specified in the JCL).
NJE	When this field contains a Y, the job has been sent for execution to a computer that is connected to a CONTROL-M computer by NJE, that is, it does not have a shared spool with CONTROL-M. Normally, do not modify the value in this field.
	BMC Software recommends that you do not purge jobs from the spool on the Remote node. However, if a job was purged from the spool on the Remote node, you must notify CONTROL-M of the event by changing the value in the <b>NJE</b> field back to ' ' (Blank). After a short time, the job status changes to Disappeared.
SEARCH COUNTER	Number of times CONTROL-M has looked for a job that is not found. (This value is displayed (as <i>n</i> ) in job status BUT NOT FOUND <i>n</i> TIMES.) When this value equals the maximum number of searches allowed, the job status changes to DISAPPEARED.
	<b>Note</b> : The default value is 10. This value can be changed by your INCONTROL administrator, using the #JNFRT parameter in the CTMPARM member in the IOA PARM library.
	You may change the value of this counter. Two instances in which this might be helpful are:
	<ul> <li>As the counter approaches the maximum number of searches allowed, set the SEARCH COUNTER back to zero if you do not want the status changed to DISAPPEARED.</li> </ul>
	■ If the search is pointless (for example, you know the job has been deleted from spool), change the SEARCH COUNTER to 99999 thereby causing a DISAPPEARED status.
LPAR	Identity of the MVS system (the logical partition) on which the job is being, or has been, executed.
NXT RUN	For rerun situations or for cyclic jobs that use the <b>INTERVAL</b> option, this field indicates the next time the job is submitted (if other submission criteria are satisfied). Format: <i>yymmdd hhmm</i> .

Table 72 Fields of the Job Scheduling Definition Zoom Screen (part 4 of 4)

Field	Description	
ON PGMST trigger	This field appears at the end of the ON PGMST line, to the right of the <b>A/O</b> field. In Figure 51 an asterisk can be seen in this field. The field is used to indicate if the ON PGMST statement was triggered Possible values are:	
	<ul> <li>'*' (Asterisk) – ON PGMST statement was triggered.</li> <li>' ' (Blank) – ON PGMST statement was not triggered.</li> </ul>	
	<b>Note:</b> If more than one ON PGMST statement has been specified:	
	■ If the statements are joined by an OR relationship, related DO actions were performed if an asterisk appears in this field for any ON PGMST statements.	
	■ If the statements are joined by an AND relationship, related DO actions were performed only if an asterisk appears in this field for all joined ON PGMST statements.	

Only specific dates (or ****, \$\$\$\$ or STAT) can be used as valid condition date references. Therefore, if symbolic date references (such as ODAT or PREV) are entered as condition date references (in the parameters IN, OUT, CODES, COND, and so on) in the job scheduling definition, the real date values are derived and displayed in the Zoom screen.

**§Restart§** The restart decision parameters (FROM, TO, CONFIRM) contain a value other than blank only if

■ the DO IFRERUN parameters have been specified in the Job Scheduling Definition screen (Screen 2)

and

■ the job was executed. § Restart §

When and if the job is restarted, these parameters are used. You can modify the value of these parameters.

The DOC command can be used to alternately display and hide the documentation (DOC lines). Documentation cannot be updated in the Zoom screen.

# **Zoom Screen for Group Entities**

An example of the Zoom screen for Group Entities is shown below.

As noted earlier, a job must be placed in the Held state before entering the Zoom screen if changes are to be saved. When a Group entity is held, changes to jobs within the Group can be saved without having to separately place a hold on each job.

All information applicable to the regular Zoom screen applies to the Group Entity Zoom screen as well. All fields in the Group Entity Zoom screen also appear in the Zoom screen for regular job scheduling definitions. For a description of the fields in the Group Entity Zoom screen, refer to the descriptions of the regular Zoom screen, the Job Scheduling Definition screen, and the Group Entity screen.

Figure 52 Zoom Screen for Group Entities

```
-----(3.Z)
COMMAND ===>
                                                           SCROLL===> CRSR
  OWNER NO4A TASKTYPE GRP
SCHDTAB SPDCRP SCHDLIB CTM.PROD.SCHED
APPL
            GROUP ACCCOUNTING
SYSTEM ID NJE NODE
JOBID ODATE 060601 ORDERID 000IH GRP MAXWAIT 00
  SCHENV
  JOBNAME
  DESC
  SET VAR
  DOCMEM ACCOUNT DOCLIB CTM.CMEM.DOC
  CONTROL
  CONTROL
FROM TIME + DAYS UNTIL TIME + DAYS
DUE OUT TIME 1314 + DAYS PRIORITY SAC CONFIRM N
  TIME ZONE:
  ON GROUP-END OK
  DO COND ACCOUNTING-OK 0909 + SHOUT WHEN TIME + DAYS TO
                                                                 URGN
 ===== >>>>>>>>> END OF JOB PARAMETERS <<<<<<<<<<<<<<<<<<<><<<<<><<<<>=====
COMMANDS: CANCEL DOC NOTE
                                                                    11.41.46
```

### **Zoom Screen Commands**

The following commands are displayed in the Zoom screen.

Table 73 Commands of the Zoom Screen

Command	Description
SAVE	Command SAVE in the Zoom screen saves changes to the screen.
DOC	Command DOC alternately displays or hides the job documentation. For more information, see "Job Documentation" on page 144.
NOTE	Command NOTE opens a note and adds it to the job order.



#### – NOTE –

When a SAVE command is processed by CONTROL-M, all SHOUT requests, even those which were already actioned, are re-queued, which may possibly cause redundant SHOUTs. To prevent specific SHOUTs from being re-queued, delete these SHOUT requests in the Zoom screen before the SAVE. These changes affect only the current job order, not the original job scheduling definition.

# **Adding or Editing a Job Order Note**

You can add, delete or change a note for the job order in the Zoom screen. For example, you might use a note to document a manual intervention in a job run. First, however, the job must be placed in Held status.

To add a note, type **NOTE** in the command line of the Zoom screen and press **Enter**. A new NOTE line is opened for entering additional notational text.

Figure 53 Adding or Editing a Job Order Note

		(	ONTROL-M	ZOOM SCI	REEN		(	
COMMAND ===	:> ·						SCROLL===>	CRSR +
MEMNAME P	PRDKPL01	MEMLIB	CTM.PROD	.JOBLIB				
		TASKTYPE			PREVENT-	NCT2	DFLT N	
SCHDIAB M		SCHDLIB	CIMP.SCH	FDULE	GROUP KP	) [		
OVERLIB	ROD				GROUP KP		T CAL	
SCHENV			SYSTEM	ΙD			NODE	
JOBNAME	JO	BID	ODAT	E 060601	ORDERID	0005C	MAXWAIT	04
	ECISION-FRO		· · · · · · ·	T0		· .	CONFIRM	N
DESC D	DAILY PRODUC	:110N - SI	ARI OF P	RODUCTIO	N GROUP K	(PL		
021 17111	AT	NAME	Т	YPF				
	PRDKPL01			.DOC				
NOTE								===
DOC								
I N	DAILY-PROD-	KPL-G0	0606					
	DB2-MAIN-FI	LE	Е					
RESOURCE		NOTE	0001	CART			0001	10.15
CUMMANDS:	CANCEL DOC	NUIE					09.	13.15

Add or edit the text in the note lines as desired. When text is added to an empty note line, a new blank note line is opened. To delete a note, delete all note lines.

When the note is as you want it, type SAVE in the Command line and press **Enter**.

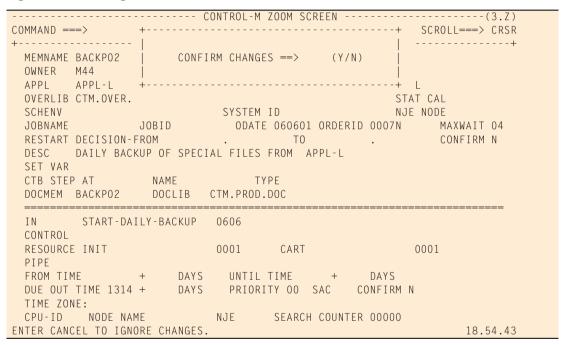
After all changes to the Zoom screen are made, free the Held job in the Active Environment screen. When a job order contains a note, an indicator is placed in the **Status** field of the Active Environment screen.

# **Exiting the Zoom Screen**

The method of exiting the Zoom screen and saving changes can be customized using the user profile.

- By default, END (**PF03/PF15**) performs a cancel, and the changes are not saved (that is, no changes are made to the job entry on the Active Jobs file). To save changes, the SAVE command must be entered.
- If customized, END (**PF03/PF15**) performs a save. In this case, the following confirmation window is displayed if changes have been made.

Figure 54 Exiting the Zoom Screen Confirmation Window



Fill in the fields of the window as follows and press **Enter**:

- Type **Y** (Yes) in the window to save the changes.
- Type N (No) in the window to cancel the changes.

To bypass the window if it is normally displayed, exit the Zoom screen as follows:

- Type **SAVE** in the Zoom screen to save changes (not available in Browse mode).
- Type **CANCEL** in the Zoom screen to cancel changes.

Upon saving changes, the status of the job becomes REQUESTED CHANGE HELD. Wait until the REQUESTED CHANGE status disappears (indicating that the CONTROL-M monitor has accepted the change), and then free the job in the Active Environment screen.

# **Confirm Scheduling Window**

If a job scheduling definition contains a value of Y in the runtime scheduling parameter CONFIRM, the job requires manual confirmation before it can be considered for submission. When such a job is placed in the Active Jobs file, it appears in the Active Environment screen with status of WAIT CONFIRMATION.

To confirm the scheduling of the job for submission, type **C** (Confirm) for the job, in the Active Environment screen. A confirmation window is then opened. The same confirmation window is opened when requesting the rerun of a job in the Active Environment screen. For the description and an example of the confirmation window, see "Confirm Rerun Window."

### **Confirm Rerun Window**

If a job scheduling definition does not contain an appropriate DO RERUN statement, or if the specified MAXRERUN limit was reached, a job is not automatically rerun if the job execution fails.

In such cases, however, rerun of the job can be manually requested by entering  ${\bf R}$  (Rerun) in the Active Environment screen.

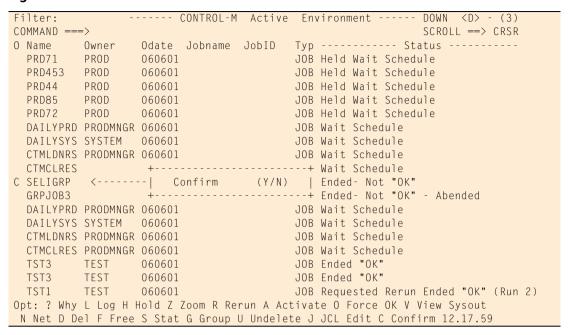
The following confirmation window is opened when either option R (Rerun) or option C (Confirm) is entered in the Active Environment screen.

#### - NOTE -



§Restart§ If CONTROL-M/Restart is available, a different window is opened for job rerun. For more information, see "§Restart§Rerun and/or Restart Window (Under CONTROL-M/Restart)" on page 224.

Figure 55 Active Environment Screen Confirm Rerun Window



When **R** (Rerun) is entered and when CONTROL-M/Restart is not installed, the following confirmation window is opened:

```
+-----+
R mem-name <----- | Rerun (Y/N) | +-----+
```

Fill in the fields of the window as follows, and press **Enter**:

**Table 74** Fields of the Confirm Rerun Window

Field	Description
Confirm or Rerun	Valid values are:
	<ul> <li>Y (Yes) – Submission or rerun of the job is requested.         The status of the job is changed to WAIT SCHEDULE, and the job is eligible for submission by CONTROL-M once all other runtime criteria are satisfied.         </li> <li>N (No) – No action is taken.         The status of the job remains unchanged and the job is not submitted.     </li> </ul>

# §Restart§ Confirm Restart Window (Under CONTROL-M/Restart)

When CONTROL-M/Restart is available, and the job scheduling definition of a job whose execution fails contains a DO IFRERUN statement, the job can be restarted. Manual intervention is required for the job restart if the job appears in the Active Environment screen with a status of WAIT CONFIRMATION (WITH RESTART). For a job requiring restart, this status appears when all the following conditions exist:

- A DO RERUN statement is defined following the DO IFRERUN statement, indicating that the job must be scheduled for restart or rerun.
- The **CONFIRM** field in the DO IFRERUN statement contains a value of Y (Yes), indicating that confirmation is required before the job is restarted.
- A MAXRERUN value greater than zero is defined in the job scheduling definition, but the number of reruns specified in this field has not yet been performed. In this case, restart can be confirmed by entering option C (Confirm) for the job.

To confirm restart or rerun for such a job, enter option C (Confirm) for the job. A restart confirmation window is then opened. The same confirmation window is opened when requesting the rerun (option R) of a restart job in the Active Environment screen. For the description and an example of the confirmation window, see the following section. "§Restart§Rerun and/or Restart Window (Under CONTROL-M/Restart)".

# §Restart§Rerun and/or Restart Window (Under CONTROL-M/Restart)

When CONTROL-M/Restart is available, and the job scheduling definition of a job whose execution fails contains a DO IFRERUN statement, the job can be restarted.

Manual intervention is required for the job restart in the following cases:

- No DO RERUN statement is defined following the DO IFRERUN statement in the job scheduling definition. In this case, the job appears in the Active Environment screen with a failed job status.
- The CONFIRM field of the DO IFRERUN statement contains a value of Y (Yes). In this case, job appears in the Active Environment screen with a failed job status.

■ No maximum number of reruns is defined in field MAXRERUN, or the maximum number reruns defined in field MAXRERUN has been performed. In this case, the job appears in the Active Environment screen with a status of ENDED NOT "OK" – RERUN NEEDED.

To manually request rerun and/or restart for such a job, enter option R (Rerun) for the job.

The following confirmation window is opened when either option R (Rerun) or option C (Confirm) is entered in the Active Environment screen for a job requiring rerun and/or restart under CONTROL-M/Restart.

Figure 56 §Restart§ Active Environment Rerun and/or Restart Confirmation Window

Filter: CC	ONTROL-M Active Environment DOWN <d> - (3)</d>
COMMAND ===>	SCROLL ==> CRSR
	Jobname JobID Typ Status
DAILYPRD PRODMNGR 110405	JOB Wait Schedule
	+(3.R)+
	Job   IEFBR14   Is to be Rerun
IOACLCND PRODMNGR 110405	
R IEFBR14 K39 110405	With Restart N (?/Y/N)   o CC
	From Step/Proc .
IEFBR14 N98A 110405	To Step/Proc .
IEFBR14 K39 110405	Recapture Abend Codes (Y/N)
IEFBR14 K39A 110405	Recapture Cond Codes (Y/N)
IEFBR14 K39A 110405	Step Adjustment (Y/N)
====== >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>	
	Flow Rerun
	(Type 'R' to change options)
	Forward Flow
	Backward Flow
	View Jobs in Flow
0.5 2 115 1 15.5 11.115.7 7 7	Restart Parm Member Name IEFBR14
, ,	++ Sysout
N NET D DELF Free S St	tat G Group U Undelete J JCL Edit C Confirm 08.36.28

Fill in the fields of the window as follows, and press **Enter**:

Table 75 §Restart§ Fields of the Active Environment Rerun and/or Restart Confirmation Window (part 1 of 4)

Field	Description
Confirm	<ul> <li>Valid values are:</li> <li>Y (Yes) – Job rerun is requested. The job is returned for possible resubmission by CONTROL-M (provided that all runtime conditions are met). The status of the job is changed according to the value of the With Restart field.</li> <li>N (No) – No action is taken. The job is not rerun. The status of the job remains unchanged.</li> </ul>

Table 75 §Restart§ Fields of the Active Environment Rerun and/or Restart Confirmation Window (part 2 of 4)

Field	Description
With Restart	This field is applicable only if Y is entered for Confirm. If N is entered for Confirm, this field is ignored. Valid values are:
	<ul> <li>Y (Yes) - When the job is rerun, it is restarted using the Restart facilities of CONTROL-M/Restart. The status of the job is changed to WAIT SCHEDULE (WITH RESTART).</li> <li>N (No) - The Restart facilities of CONTROL-M/Restart are not used. The status of the job is changed to WAIT SCHEDULE.</li> <li>? - Opens the Restart Step List window, which contains a list of the job's steps. This list can then be used for specifying From Step and To Step values. For more information, see "Step List Window" on page 233.</li> </ul>
From Step/Proc	The <i>pgmstep</i> (and optionally <i>procstep</i> ) names at which the restart of the job is to be attempted.
To Step/Proc	The <i>pgmstep</i> (and optionally <i>procstep</i> ) names at which the restarted job terminates processing. Optional.
	The From Step/Proc and To Step/Proc fields display from and to step values specified in the DO IFRERUN statement. These values may be modified.
	If a From Step/Proc value is specified, and the To Step/Proc field is blank, the job is rerun up to and including the last step.
Recapture Abend Codes	Whether to enable abend code recapture. This field is applicable only if Y is entered for WITH RESTART. (If N is entered for WITH RESTART, this field is ignored.) Valid values are:
	<ul> <li>Y (Yes) - Automatic abend code recapture is performed.</li> <li>N (No) - Abend code recapture is prevented.</li> <li>' (Blank) - The job member or the \$DEFAULT member in the CTR.PARM library is used. If the \$DEFAULT member is not found, the CONTROL-M/Restart default is used to perform the recapture.</li> </ul>
	<b>Note</b> : If ordering a restart of a job from a step after an abended step, type N in this field. If not, only steps that specify the JCL parameters COND=ONLY or COND=EVEN run during restart.
Recapture Cond Codes	This field is applicable only if Y is entered for With Restart. (If N is entered for With Restart, this field is ignored.) Valid values are:
	<ul> <li>Y (Yes) - Automatic condition code recapture is performed</li> <li>N (No) - Condition code recapture is prevented</li> <li>' (Blank) - The job member or the \$DEFAULT member in the CTR.PARM library is used. If the \$DEFAULT member is not found, the CONTROL-M/Restart default is used to perform the recapture.</li> </ul>

Table 75 §Restart§ Fields of the Active Environment Rerun and/or Restart Confirmation Window (part 3 of 4)

Field	Description
Step Adjustment	This field is applicable only if Y is entered for With Restart. (If N is entered for With Restart, this field is ignored.) Valid values are:  ■ Y (Yes) – Automatic step adjustment is performed. ■ N (No) – Step adjustment is prevented. ■ ' ' (Blank) – The job member or the \$DEFAULT member in the CTR.PARM library is used. If the \$DEFAULT member is not
	found, the CONTROL-M/Restart default is used to perform the step adjustment.
	for Recapture Abend Codes, Recapture Cond Codes and Step all other parameter specifications and the default. They apply to the
Flow Rerun	Whether to open the Flow Rerun Options window. Type <b>R</b> in this field to open the window.
Forward Flow	This is a display only field that indicates whether to enable the restart of a successive series of jobs, beginning with the selected job:  Y (Yes) – The job is restarted along with all its successor jobs.  N (No) – Only the selected job is restarted.
Backward Flow	This is a display only field that indicates whether to enable the restart of a preceding series of jobs, ending with the selected job:
	<ul> <li>Y (Yes) - The job is restarted along with all its predecessor jobs.</li> <li>N (No) - Only the selected job is restarted.</li> </ul>

Table 75 §Restart§ Fields of the Active Environment Rerun and/or Restart Confirmation Window (part 4 of 4)

Field	Description
View Jobs in Flow	This is a display only field that indicates whether to generate a Rerun Flow Job List. This list enables the user to select specific jobs in the selected flow or flows to be restarted. For more information, see "Rerun Flow Job List Window" on page 230.  Y (Yes) – The Flow Rerun Job List screen is displayed. The list
	<ul> <li>T (Tes) - The Flow Refull Job List screen is displayed. The list contains:         <ul> <li>The job's successors (if Y was specified in Forward Flow)</li> <li>The jobs's predecessors (if Y was specified in Backward Flow)</li> <li>The jobs's successors and predecessors (if Y was specified in both fields).</li> </ul> </li> <li>N (No) - The Rerun Flow Job List is not displayed.</li> </ul>
	If the job is a group entity, and Y was not specified in either the Forward Flow or Backward Flow fields, the generated list contains the jobs in the group. (If Y was not specified in either of the fields, and the job is not a group entity, then this value is ignored and no list is displayed.)
Restart Parm Member Name	Name of the member that contains control parameters for the job restart. The specified value must be a valid member name of 1 through 8 characters. The default value, displayed in the window, is the member that contains the JCL of the job. This member is either the value in the MEMNAME field of the Zoom screen, or the NAME field of the Active Environment screen.

Note the following points about From Step/Proc and To Step/Proc values:

- Pgmstep name can be any specific program step name or \$FIRST. \$FIRST resolves to the first step of the job if procstep name is blank. Otherwise, \$FIRST resolves to the first step in the procedure identified by procstep.
- \$ABEND and \$EXERR are not recognized by CONTROL-M/Restart and must not be specified as restart steps in this window. (\$ABEND and \$EXERR are valid only in job scheduling definitions.)
- If specifying a procstep name when there are nested procedures, specify the procstep name of the innermost procedure in which the program is included.
- Entering \$FIRST in the first From Step/Proc field followed by \$CLEANUP in the adjacent (second) From Step/Proc field reruns the job for Cleanup (that is, run the CONTROL-M/Restart cleanup step and flushes the job). All other parameters entered in the Restart window are ignored.

### — NO



AutoEdit resolution is performed at time of cleanup job submission. For example, if a job with AutoEdit date variable %%DATE is submitted for cleanup the day after the original run, the resolution of the variable during cleanup varies from that of the original run.

The RERUN request and, in CONTROL-M/Restart, the RESTART decision are recorded in the IOA Log file. If the CONTROL-M monitor is active, the RERUN request is accepted after a few seconds.

# **Flow Rerun Options Window**

This section explains the Flow Rerun Options window. Figure 57 shows an example of the Flow Rerun Options window. Table 76 describes the fields on the Flow Rerun Options window.

Figure 57 §Restart§ Flow Rerun Options Window

Filter: CON	TROL-M Active Environment DOWN <d> - (3)</d>
COMMAND ===>	SCROLL ==> CRSR
O Name Owner Odate Jo	bname JobID Typ Status
	JOB Wait Schedule
	(3.R)+
	Job IEFBR14 Is to be Rerun
· ·	Please Confirm (Y/N)
R IEFBR14 K39 110405	With Restart N (?/Y/N) o CC
7550014 110405	From Step/Proc .
IEFBR14 N98A 110405	To Step/Proc .
IEFBR14 K39 110405  +	+
1.1	Forward Flow _ (Y/N)
1.1	Backward Flow _ (Y/N)
	View Jobs in Flow _ (Y/N)     < ======
+       	(Type 'R' to change options) Forward Flow Backward Flow View Jobs in Flow
	Restart Parm Member Name IEFBR14
Opt: ? Why L Log H Hold Z Z+-	+ Sysout
	t G Group U Undelete J JCL Edit C Confirm 08.36.28

Table 76 §Restart§ Fields of the Flow Rerun Options Window

Field	Description
Forward Flow	Whether to enable the restart of a successive series of jobs, beginning with the selected job:
	<ul> <li>Y (Yes) - The job is restarted along with all its successor jobs.</li> <li>N (No) - Only the selected job is restarted.</li> </ul>
Backward Flow	Whether to enable the restart of a preceding series of jobs, ending with the selected job:
	<ul> <li>Y (Yes) – The job is restarted along with all its predecessor jobs.</li> <li>N (No) – Only the selected job is restarted.</li> </ul>
View Jobs in Flow	Whether to generate a Rerun Flow Job List. This list enables the user to select specific jobs in the selected flow or flows to be restarted. For more information, see "Rerun Flow Job List Window" on page 230.
	■ Y (Yes) – The Flow Rerun Job List screen is displayed. The list contains:
	<ul> <li>The job's successors (if Y was specified in Forward Flow)</li> <li>The job's predecessors (if Y was specified in Backward Flow)</li> <li>The jobs's successors and predecessors (if Y was specified in both fields).</li> <li>N (No) - The Rerun Flow Job List is not displayed.</li> </ul>
	If the job is a group entity, and Y was not specified in either the Forward Flow or Backward Flow fields, the generated list contains the jobs in the group. (If Y was not specified in either of the fields, and the job is not a group entity, then this value is ignored and no list is displayed.)

# **Rerun Flow Job List Window**

A list of the jobs that may be restarted from a job flow rerun request can be displayed from the Active Environment screen, by selecting the View Jobs in Flow option in the Rerun and/or Restart Confirmation window. The user can then select specific jobs from this list to be restarted or not, by using the I (Include) and E (Exclude) options.

Figure 58 Rerun Flow Job List Window

```
----- RERUN FLOW --- JOB LIST
                                             ----- UP \langle F \rangle - (3)
Filter:
COMMAND ===>
                                                  SCROLL ==> CRSR
0 Level ---- N a m e ---- Rerun?
                                               ----- Status -----
 -1 ADDRESOR
                                      Ended "OK"
                                      Ended "OK"
 -1 SOURCERN
                       YES
 - 2
                                      Ended "OK"
     CEDOLXRX
                       YES
                       YES
 -1 CTDORDER
                                      Ended "OK"
                       YES
 -1 QD61P38
                                     Ended "OK"
Commands: OPt EXclude INclude CANcel
       OPt command toggles between Commands and Options display 15.01.32
```

The following fields are displayed for each job in the list:

**Table 77** Fields of the Rerun Flow Job List Window

Field	Description	
O(ption)	Field for requesting options to be activated.	
Level	Successor or predecessor level relative to the selected job. The current job is indicated by>. Predecessor jobs are indicated by a minus sign and successor jobs are indicated by a plus sign. Jobs that have several paths to or from the selected job appear with the shortest possible route as their level number.	
Name	Name of the member containing the JCL of the job, or name of the started task.	
Rerun?	Whether the job is to be rerun. Valid values are YES and NO.  When the list is first generated, this field is set to YES for all jobs.	
Status	Job (task) status. A complete list of job statuses is found in "Job Statuses" on page 185. One of the listed statuses appears only in this window:	
	ON HST FILE – Job is currently in the History file. If the job is included in the flow of jobs being rerun, it will be restored to the Active Jobs file before being rerun.	

To specify rerun/restart options for individual jobs:

- Type I (Include) next to specific jobs to rerun/restart them. The status for the selected job(s) is changed to YES.
- Type E (Exclude) to prevent specific jobs from being rerun/restarted. The status for the selected job(s) is changed to NO.

To specify rerun/restart options for the entire Rerun Flow Job List at once:

- Type IN (Include) on the command line to rerun/restart all jobs in the list. The status for all jobs are changed to YES.
- Type EX (Exclude) on the command line to prevent all jobs in the list from being rerun/restarted. The status for all jobs are changed to NO.

To rerun all the included jobs and exit the window, type RUN on the command line. To exit the window without rerunning any of the included jobs, type CANCEL on the command line. Pressing the END key (**PF03/PF15**) causes all the included jobs to be rerun if profile variable RESWPF3 is set to N; otherwise, it exits the window without rerunning any of the included jobs.

If you choose to rerun all the included jobs, either by typing RUN or by pressing the END key (if profile variable RESWPF3 is set to N) a confirmation window will be displayed.

Figure 59 Rerun Flow Job List Confirmation Window

```
----- RERUN FLOW --- JOB LIST
Filter:
                                                                               ----- UP 〈F〉 - (3)
                                                                                        SCROLL ==> CRSR
COMMAND ===>

      -- Rerun?

      YES
      Ended "OK"

      YES
      Ended "OK"

O Level ----- N a m e ----- Rerun?
                                                                                   ----- Status -----
                                                               Ended "OK"
  -1 ADDRESOR
   -1 SOURCERN
  -2 CEDOLXRX
-1 CTDORDER
 -1 QD61P38
--> QD61P38
------>>>>>>>>>>>>>
                                         Bottom of Jobs List <<<<<<<< =======
                 | Type 'R' to rerun flow, 'C' to cancel:
Commands: OPt EXclude INclude CANcel
             OPt command toggles between Commands and Options display 15.01.32
```

Type one of the following values in the confirmation window:

- R to rerun all the included jobs
- C to exit the window without rerunning jobs

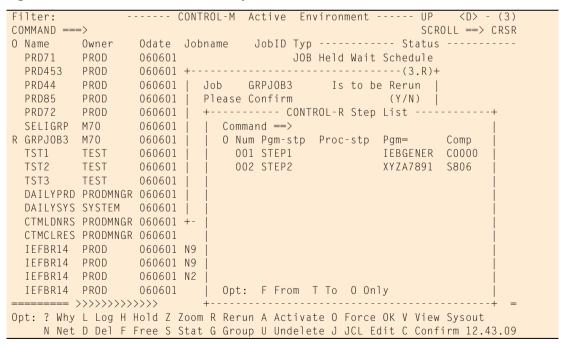
The order in which the Rerun Flow jobs are displayed can be controlled by the RSORT command. Typing RSORT FLOW on the command line causes the jobs to be displayed in the order of the flow, that is, with predecessor jobs before their successors. Typing RSORT NAME on the command line causes the jobs to be displayed in alphabetical order.

# **Step List Window**

A list of the job steps from the previous job run, with the completion codes of each step, can be displayed from the Restart window in the Active Environment screen. Steps from this list can then be selected as From Step/Proc and/or To Step/Proc values in the Restart window.

To display the list of job steps, type a ? symbol in the With Restart field of the Restart window, and press **Enter**. The Step List window, below, is opened. This "window within a window" contains the list of job steps.

Figure 60 Rerun and/or Restart Step List Window



To locate a specific step, type the LOCATE command in the Command field of the CONTROL-M/Restart Stop List window and press **Enter**. The format of the command is:

#### LOCATE stepname

Steps selected in the Step List window are displayed in the appropriate field of the Restart window. To select steps, type the appropriate selection values in the Option (O) field by the step names. Valid selection values are shown in Table 78.

Table 78 Options of the Rerun and/or Restart Step List Window

Option	Description
F (From)	Restart begins at the indicated step. The indicated step becomes the From Step/Proc parameter.
T (To)	Restart ends at the indicated step. The indicated step becomes the To Step/Proc parameter.
O (Only)	Restart begins and ends at the indicated step. The indicated step becomes the From Step/Proc and To Step/Proc parameter. This value cannot be specified with an F or a T value.

#### NOTE



If a step cannot be used as a From Step/Proc and/or To Step/Proc for restart, the Option field is protected, and an option cannot be entered, for that step.

Pressing the END key (PF03/PF15) closes the Step List window and automatically updates the From Step/Proc and To Step/Proc fields of the Restart window with the appropriate steps.

Typing the command RESET, or pressing the RESET key (**PF04**), closes the Step List window without updating the Restart window.

# §Restart§Job Order Execution History Screen

The Job Order Execution History screen is displayed when option V (View Sysout) is entered on the Active Environment screen. The Job Order Execution History screen lists all runs of the job and displays relevant execution information for each run.

This option is used for jobs that have executed at least once and whose SYSDATA has been automatically archived by the CONTROL-M monitor as part of CONTROL-M/Restart processing (Auto-Archive=Y). The V option does not support the viewing of output on the spool or output that has been archived by a CONTROL-M SYSOUT F function.)

Figure 61 §Restart§ Job Order Execution History Screen

```
COMMAND ===>

COMMAND ==>

COMMAND ===>

COMMAND ==>

COMMAND ===>

COMM
```

By default, older data are displayed before more recent data (that is, in FIFO order – first in, first out), so that the first run of the job is shown first. If, however, the user profile has been customized accordingly, data is displayed in LIFO order (last in, first out).

The Job Order Execution History screen is pre-configured to the D (Default) display type. Additional display types may be defined by the INCONTROL administrator. To display a different display type on the screen, type command DISPLAY *x* (abbreviated DI *x*) where *x* is the identifying letter of the display type (such as DI D).

To return to the Active Environment screen, press the END key (PF03/PF15).

# §Restart§ Format of the Job Order Execution History Screen

The following information about the job is displayed in the Default display type of the Job Order Execution History screen.

Table 79 §Restart§ Default Display Type Fields of Job Order Execution History Screen

Field	Description	
MEMNAME	Name of the member containing the job's JCL.	
OWNER	User ID of the owner of the job.	
ORDERID	Job order ID.	
ODATE	Original scheduling date of the job.	

For each execution of the job, the following information is displayed:

Table 80 §Restart§ Fields in the Job Order Execution History Screen

Field	Description	
0	Option field.	
JOBNAME	Job name.	
JOBID	JES job number.	
DATE	Execution date of the job.	
START	Start time of the job execution (format hh:mm).	
ELAPSED	Total elapsed time of the job execution (format mmmm.nn – where mmmm is minutes, and nn is hundredths of minutes).	
PAGES	Number of pages in the sysout.	
MAX RC	Highest return code of the job execution.	
STATUS	Status assigned to the job by CONTROL-M, based on execution results.	

# **§Restart§ Displaying Job Sysout**

Job execution sysout, which is displayed in the Sysout Viewing screen, can be requested from the Job Order Execution History screen in the following ways:

- To display job sysout for specific executions of the job, type the option S (Select) in the OPTION field of the selected executions and press **Enter**.
- To display job sysout for all executions of the job, type the command VIEWALL (abbreviated V) in the COMMAND field and press **Enter**.

# **§Restart§ Sysout Viewing Screen**

The Sysout Viewing screen is displayed when the option S (Select) is entered for specific job executions, or when the command VIEWALL is entered, in the Job Order Execution History screen.

This screen displays the job execution sysout, as follows:

- If the option S (Select) was typed in the Job Order Execution History screen for specific executions of the job, the sysout for those executions are displayed.
- If the command VIEWALL was entered in the Job Order Execution History screen, the sysout for all executions is displayed.

Figure 62 §Restart§ Sysout Viewing Screen

```
CONTROL-M/CONTROL-R SYSOUT VIEWING -----
                                                            1 OF
                                                                     12
COMMAND ===>
                                                          SCROLL ===> CRSR
MEMNAME PRDKPL01 OWNER M22
                                JOBNAME PRDKPL01
                                                    ODATE 060601
JES2 JOB LOG -- SYSTEM FDSF -- NO
18.31.20 JOB 8666 $HASP373 PRDKPL01 STARTED - INIT 1 - CLASS A - SYS FDSF
18.31.20 JOB 8666 IEF403I PRDKPL01 - STARTED - TIME=18.31.20
18.35.21 JOB 8666 PRDKPL01.STEP01 .#01; - COMPLETION CODE=0000
18.39.22 JOB 8666 PRDKPL01.STEP01A .#02; - COMPLETION CODE=0000
18.42.22 JOB 8666 PRDKPL01.STEP02 .#03; - COMPLETION CODE=0000
                                .#04; - COMPLETION CODE=0000
18.50.23 JOB 8666 PRDKPL01.STEP03
                 IEF450I PRDKPL01 STEP04 - ABEND SOC4 U0000 - TIME=18.51.25
18.51.25 JOB 8666
                 PRDKPL01.STEP04 .#05; - COMPLETION CODE=SOOC4 - ABENDED排排
PRDKPL01.STEP05 .#06; - COMPLETION CODE=NOT RUN
18.51.25 JOB 8666
18.51.25 JOB 8666 PRDKPL01.STEP05
18.51.25 JOB 8666 PRDKPLO1.STEP06 .#07; - COMPLETION CODE=NOT RUN
18.51.25 JOB 8666 PRDKPL01.STEP07
                                .#08; - COMPLETION CODE=NOT RUN
18.51.26 JOB 8666 PRDKPL01.STEP08 .#09; - COMPLETION CODE=NOT RUN
18.51.26 JOB 8666 PRDKPL01.STEP09 .#10; - COMPLETION CODE=NOT RUN
18.51.26 JOB 8666 PRDKPL01.STEP10 .#11; - COMPLETION CODE=NOT RUN
18.51.26 JOB 8666 $HASP395 PRDKPL01 ENDED
----- JES2 JOB STATISTICS -----
COMMANDS: LEFT, RIGHT, FIND str, FIND str PREV, N n, P n, END
                                                                18.56.48
```

Job orders are displayed in the same order (LIFO/FIFO) in this screen as in the Job Order Execution History screen.

To return to the Job Order Execution History screen, press END (PF03/PF15).

The following commands are supported:

Table 81 §Restart§ Commands of the Sysout Viewing Screen

Command	Description	
LEFT	Shift display to the left.	
RIGHT	Shift display to the right.	
<b>Note</b> : Terminals with 132-character lines display the entire data line. Therefore, LEFT and RIGHT do not affect the display on those terminals.		
FIND str	Find next occurrence of the string.	
FIND str PREV	Find previous occurrence of the string.	
NEXT n	Scroll forward $n$ number of print pages (can be abbreviated N $n$ ).	
PREV n	Scroll backward <i>n</i> number of print pages (can be abbreviated P <i>n</i> ).	
END	Exit the screen.	

For color terminals, display colors for the sysout are defined in the user profile. If you want to change the default colors, see your INCONTROL administrator.

### **Statistics Screen**

The Statistics screen displays the most current run statistics for a particular job. The screen is displayed when any of the following actions is performed:

- Option S (STAT) is typed next to the jobname in the Active Environment screen.
- Option T (JOBSTAT) is typed next to the job name in the Job List screen.
- Command JOBSTAT is typed in Command field of the Job Scheduling Definition screen or the Active Environment screen.

A separate set of statistics is collected for each group on each computer in which the job is run. Statistics for a job are retained for a maximum of 200 executions in each group on each computer.

Figure 63 Active Environment Statistics Screen

BR14 STATISTICS	(3 5)
	OLL===> CRSR
JOBNAME JOBID STATUS STRT DATE/TIME END DATE/TIME ELAPSED	CPU
ELAPSED AVERAGE: 0.01 MAX: 0.01 MEDIAN: 0.01 STD:	0.01
SYSID: 1 SMFID: ESA1 PERIOD: A AVERAGE: 0.00	
NEWJOB 0016156 04/07/06 14:13 04/07/06 14:13 0.00	
BR14A 0016157 04/07/06 14:13 04/07/06 14:13 0.00	0:00.01
BR14B 0016144 04/07/06 13:55 04/07/06 13:55 0.00	0:00.00
0016146 04/07/06 13:55 04/07/06 13:55 0.00	0:00.01
ABNDJOB 0016133 NOTOK 04/07/06 13:41 04/07/06 13:41 0.00	
IEFBR14 0016134 DUMMY 04/07/06 13:41 04/07/06 13:41 0.00	0:00.01
0016135 04/07/06 13:41 04/07/06 13:41 0.00	0:00.01
JCLERR 0016131 NOTOK 04/07/06 13:39 04/07/06 13:39 0.00	0:00.00
ELAPSED AVERAGE: 0.00 MAX: 0.00 MEDIAN: 0.00 STD:	
SYSID: 3 SMFID: OS33 PERIOD: AVERAGE: 0.00	
0016167 04/07/06 14:41 04/07/06 14:41 0.00	0:00.01
MYJOB 0016168 04/07/06 14:41 04/07/06 14:41 0.00	0:00.01
YOURJOB 0016155 04/07/06 14:13 04/07/06 14:13 0.00	
0016146 04/07/06 13:55 04/07/06 13:55 0.00	
===== >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>	<b>⟨⟨⟨⟨⟨ ====</b>
PRESS END PFK TO RETURN PRESS PF11 TO SCROLL RIGHT	14.28.25

To return to the Active Environment screen, press the END key (PF03/PF15).

#### **NOTE**



Update of the Statistics file is performed by the CTMJSA utility, which must be scheduled periodically. The CTMJSA utility is described in the *INCONTROL* for z/OS Utilities Guide.

#### WARNING



If statistics that exist for a job are not displayed, refresh the display by entering the REFRESH command (PF04/PF16).

### **Fields of the Statistics Screen**

For each computer with statistics on the job, an Average Statistics line is displayed followed by individual job or Group Entity statistics for each execution. Individual job and Group Entity statistics are listed in descending chronological order, with the most recently ended job at the top.

#### **Average Statistics Lines**

The second line following the report heading contains:

- SYSID of the computer for which statistics are calculated
- SMF ID corresponding to the SYSID
- statistical PERiod indicator specified in the job definition's STAT CAL
- average CPU time
- average SRB time

The first line following the header contains the following statistics for the jobs which are grouped by the SYSID and PERiod specified above:

- AVERAGE ELAPSED time
- MAXimum elapsed time
- MEDIAN elapsed time
- STandard Deviation elapsed time (see utility CTMJSA in the *INCONTROL for z/OS Utilities Guide* for details on calculating standard deviation)

#### NOTE



The SYSID consists of one or two digits, left-justified.

#### **Individual Execution Statistics**

Table 82 Statistics Screen Individual Execution Statistics (part 1 of 2)

Field	Description	
JOBID	Job number under JES. Blank for DUMMY jobs (and some NOTOK jobs).	
JOBNAME	Job name from the JCL JOB statement. Blank for DUMMY jobs.	
STATUS	Job's status. Valid values are:	
	■ ''(Blank) – job ended OK (or Forced OK)	
	<ul> <li>NOTOK – job ended NOTOK (or required a RERUN)</li> <li>DUMMY – dummy job</li> </ul>	

Table 82 Statistics Screen Individual Execution Statistics (part 2 of 2)

Field	Description	
STRT DATE/TIME	Date and time the job began executing. Date format: mmddyy, yymmdd or ddmmyy depending on site standard. Time format: hh:mm, where hh=hours and mm=minutes.	
END DATE/TIME	Date and time the job finished executing. Same format as STRT DATE/TIME.	
ELAPSED	Elapsed runtime. Format: mmmm.ss, where mmmm is minutes and ss is seconds.	
CPU	CPU time used. Format mmmm:ss.nn, where mmmm is minutes, ss is seconds and nn is hundredths of seconds.	
SRB	SRB (System Request Block) time used. Format: mmmm:ss.nn, where mmmm is minutes, ss is seconds and nn is hundredths of seconds.	
USER DATA	Optionally supplied data from the user data area in the CONTROL-M Statistics file (edited by user Exit CTMX013).	

### **Group Entity Execution Statistics**

Fields of the Group Entity Execution statistics have different meanings than corresponding fields of the Individual Execution statistics.

**Table 83 Statistics Screen Group Entity Execution Statistics** 

Field	Description	
JOBID	Order ID of the group entity.	
JOBNAME	Blank	
STATUS	Status of the jobs in the group. Valid values are:	
	<ul> <li>' ' (Blank) – all jobs in the group ended OK (or Forced OK)</li> <li>NOTOK – at least one job in the group did not end OK (or required a RERUN)</li> </ul>	
STRT DATE/TIME	Date and time the group began executing. Date format: mmdd or ddmm depending on site standard. Time format: hh:mm (where hh=hours and mm=minutes).	
END DATE/TIME	Date and time the group finished executing. Same format as STRT DATE/TIME.	
ELAPSED	Elapsed time from the time the first job in the group began executing until the time the last job in the group finished executing.	
CPU	0	
SRB	0	
USER DATA	Blank	

To view the entire screen displayed in Figure 63, scroll to the right (using PF11/PF23).

#### **Tape Device Usage Statistics**

If the AUTOTAPE parameter in the CTMPARM member in the IOA PARM library is set to Yes, tape device usage information is accumulated for every CONTROL-M job execution that ended OK. This information is used by the Automatic Tape Adjustment facility to automatically allocate the appropriate number of tape drives for a job at job order time. This allocated value overrides any specified tape device usage value in the RESOURCE parameter. For more information, see the discussion of using the automatic tape adjustment facility in the *INCONTROL for z/OS Administrator Guide*.

This information (shown below) can be displayed by scrolling to the right of the Statistics screen (using **PF11/PF23**):

Figure 64 Tape Device Usage Statistics

```
DEVICES USED
       STRT DATE
JOBID
AVERAGE: SYSID:
0239
       05/02/01
                    TAPE=1; CARTRIDGE=1;
                    TAPE=1; CARTRIDGE=1;
0643
       06/02/01
0399
       07/02/01
                    TAPE=1:CARTRIDGE=1:
2141
       12/02/01
                    TAPE=1; CARTRIDGE=1;
0493
       13/02/01
                    TAPE=1; CARTRIDGE=1;
```

The tape usage information consists of fields JOBID and START date (from the Statistics screen) so that tape usage of a specific job execution can be easily identified, and an additional field, DEVICES USED, which is described below.

The DEVICES USED field contains tape device types and number of devices of each type that were used by the job. This field has the following format:

```
devtype1=quant1;devtype2=quant2;...devtypex=quantx;
```

#### where:

- *devtype* A tape device type used by the job.
- *quant* The number of tape devices of the specified type used by the job.

The maximum quantity of units per device type for any job that can be accommodated by the Statistics file is 15. Any actual quantity exceeding 15 will be stored and displayed as 15.

Tape device types are displayed in the order specified by the INCONTROL administrator in the UNITDEF member of the CONTROL-M PARM library.

If the tape device usage information occupies more than the visible screen, scroll again to the right (using PF11 or PF23) to view additional device usage information. The maximum length of tape device usage data is 255 characters.

# Job Dependency Network Screen

The Job Dependency Network screen is displayed when option N (Network) is entered on the Active Environment screen. The Job Dependency Network screen displays all the predecessor and successor jobs (including eventual successors and predecessors) for the selected job.

Job dependencies are determined according the prerequisite IN and OUT conditions of the job. DO COND statements are not used for this purpose because the dependencies they create are conditional rather than constant.

The Job Dependency Network screen is a special case of the Active Environment screen, and therefore contains most of the same features. A filter that is active in the Active Environment screen also remains active in the Job Dependency Network screen (and vice versa).

Jobs are listed in job flow order (that is, level) relative to the selected job. The selected job is indicated by level 0. Predecessor jobs are indicated by a minus sign and successor jobs are indicated by a plus sign.

The network of jobs is maintained by the CONTROL-M monitor, and is refreshed only by request. The REFRESH Command is described in "Commands of the Job Dependency Network Screen" on page 244. The time of the last network refresh is displayed on the top line of the Job Dependency Network screen.

To return to the Active Environment screen, press the END key (PF03/PF15).

Features of the Job Dependency Network screen that have already been described earlier for the Active Environment screen are not described below. Refer to the Active Environment screen for information about those features. However, note the following points:

- The RBAL command of the Active Environment screen is not supported in the Job Dependency Network screen. This difference is also reflected in the Primary Bottom lines of the two screens.
- The N (Network) display type is specifically oriented to the Job Dependency Network screen, and is therefore described in the following section. It is also available in the Active Environment screen.

For a description of the fields in the D (Default) display type and the A (All Info) display type, see "Format of the Active Environment Screen" on page 170.

# **Format of Job Dependency Network Information**

### **Display Type N (Network)**

The Network display type is intended for use by the INCONTROL administrator and operations personnel. Basic information is displayed for each job.

Figure 65 Job Dependency Network Display Type N (Network)

Ī	Filter: DEFAULT CO	NTROL-M NETWORK OF BG	CBHK6	UP <n> - (3)</n>
	COMMAND ===>			SCROLL ==> CRSR
	O Level N a m e			
		1206 1209 0003		
		1209 1212 0003		WAIT SCHEDULE
		1212 1215 0003		
	-5 CHECKFL2		9	
	-4 CHECKFL3		9	
	-4 LOGLIST		9	
	-4 FLOWCHK		9	
	-3 MAINTST		9	
		1230 1233 0003	9	WAIT SCHEDULE
	-1 FLOWPRT		9	
	> RGL1 +1 RGL2	1236 1239 0003	9	WAIT SCHEDULE
			9	
	+2 RGLCHK		9	WAIT SCHEDULE
	+3 RGL3		9	
	+4 DELCHK		9	
	+5 DELLOG		2	
		1254 1257 0003		
		1257 1300 0003		
	====== >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>			
	Commands: OPt DIsplay Show H			
	OPt command toggle	s between Commands and	Optio	ns display 15.15.48

Table 84 Fields of the Job Dependency Network Display Type N (Network) (part 1 of 2)

Field	Description
Filter	Name of the currently active screen filter. For more information, see "Filtering the Active Environment Screen Display" on page 189.
	IOA profile variable SACT3NFL, described in the <i>INCONTROL</i> for <i>z/OS Administrator Guide</i> , determines whether the filter in effect in the Active Environment screen also filters this display.
CONTROL-M status	Indicator of whether the CONTROL-M monitor is UP, DOWN or SUSP (suspended).
Display Type Indicator	Indicator of the currently used display type, for example, N for the Network display type.
The following are displayed for each job:	
O(ption)	Field for requesting options to be activated.

Table 84 Fields of the Job Dependency Network Display Type N (Network) (part 2 of 2)

Field	Description	
Level	Successor or predecessor level relative to the selected job. The current job is indicated by>. Predecessor jobs are indicated by a minus sign and successor jobs are indicated by a plus sign. Jobs that have several paths to or from the selected job appear with the shortest possible route as their level number.	
Name	Name of the member containing the job's JCL, or name of the started task.	
DueIn	Due in date and time. Date and time by which the job must be submitted.	
DueOut	Due out date and time. Date and time by which the job must finish executing.	
Elaps	Elapse time. Expected time (in minutes) for the job to execute.	
Late	<ul> <li>Indication that a job is late. Possible values:</li> <li>■ X - Actual execution has not completed within the expected execution time. Also indicates that SHOUT WHEN EXECTIME was issued.</li> <li>■ I - Job was not submitted in time. Also indicates that SHOUT WHEN LATESUB was issued.</li> <li>■ O - Job is late. Also indicates that SHOUT WHEN LATE was issued.</li> </ul>	
Prio	CONTROL-M priority of the job.	
Res	Indicator that the job accesses Quantitative resources. Valid values are:    ' ' (Blank) – Quantitative resources are not accessed   Y (Yes) – Quantitative resources are accessed	
Status	Job (task) status.	

# **Commands of the Job Dependency Network Screen**

Except for command REFRESH, detailed descriptions of the Job Dependency Network screen commands can be found in "Commands of the Active Environment Screen" on page 174.

Command REFRESH is described here in detail because this command is most relevant to the Job Dependency Network screen.

Command REFRESH causes the CONTROL-M monitor to recalculate job dependencies. During the refresh, that is, from the time the refresh is initiated until the refresh is completed, a special status message is displayed at the top of the screen. The format of the command is:

REFRESH parm

where parm is the type of refresh to be performed.

The following parameters may be entered with the REFRESH command:

**Table 85** Parameter of the REFRESH Command

Parameter	Description
NET	Update the list of dependent jobs in the Job Dependency Network screen. As soon as possible, the monitor recalculates logical dependencies for all job orders currently present in the Active Jobs file and updates the Job Dependency Network screen. Default.
DEADLINE	Adjust DUE OUT times, if necessary, for all job orders in the Active Jobs file that are not Held. For an explanation of the method used to recalculate DUE OUT time, see "Automatic Job Flow Adjustment" on page 74.
PROPAGATE	Check and adjust the priority of predecessor jobs. For more information, see "Automatic Job Flow Adjustment" on page 74.
ALL	Activates the processes described above (NET, DEADLINE and PROPAGATE) simultaneously in the CONTROL-M monitor.

# **History Environment Screen**

Jobs can be automatically moved from the Active Jobs file to the History Jobs file during the subsequent New Day processing.

Jobs in the History Jobs file can be displayed in the History Environment screen.

The History Environment screen is a special case of the Active Environment screen. It is displayed when command HISTORY is typed in the Command field in the Active Environment screen.

#### Figure 66 History Environment Screen

```
Filter: DEFAULT
                      ----- CONTROL-M History Environment ----- DOWN
COMMAND ===>
                                                                                 SCROLL ==> CRSR
O Name
              Owner
                         Odate Jobname JobID
                                                       Typ ----- Status -----
  DAILYSYS SYSTEM 060601
                                                       JOB Wait Schedule
   CTMLDNRS PRODMNGR 060601
                                                       JOB Wait Schedule
   CTMCLRES PRODMNGR 060601
                                                       JOB Wait Schedule
              PRODMNGR 060601 PRDGEN1 /17048
                                                       JOB Ended "OK"
  GEN2
              PRODMNGR 060601 PRDGEN2 /17049
                                                       JOB Ended "OK"
  GEN3
              PRODMNGR 060601 PRDGEN3 /17050
                                                       JOB Ended "OK"
  GEN4
GEN5
              PRODMNGR 060601 PRDGEN4 /17051
                                                       JOB Ended "OK"
             PRODMNGR 060601 PRDGEN5 /17053
                                                       JOB Ended "OK'
  TPCICS47 TP05 060601 TPCICS47/18081 TPCICS12 TP01 060601 TPCICS12/18082
                                                       JOB Ended "OK'
                                                       JOB Ended "OK"
              PRODMNGR 060601 PRDGEN1 /18084
                                                       JOB Ended "OK"
  GEN1
          PRODMNGR 060601 PRDGEN2 /18085
                                                       JOB Ended "OK"
  GEN2
  TPCICS05 TP05 060601 TPCICS05/18090 JOB Ended "OK"
Y01ACCB ACCT 060601 Y01ACCB /19053 JOB Ended "OK"
Y01ACCC ACCT 060601 Y01ACCB /19150 JOB Ended "OK"
Y01ACCD ACCT 060601 Y01ACCB /19230 JOB Ended "OK"
Y01ACCE ACCT 060601 Y01ACCB /19232 JOB Ended "OK"
  Y01ACCF ACCT 060601 Y01ACCB /19233 J0B Ended "OK" Y01ACCG ACCT 060601 Y01ACCB /19501 J0B Ended "OK"
Commands: OPt DIsplay Show HIstory RBal REFresh Auto Jobstat SHPF Note Table
             OPt command toggles between Commands and Options display
```

Because the History Environment screen is a special case of the Active Environment screen, the features of the two screens are almost identical, and are described in "Active Environment Screen" on page 167. Differences between the screens are as follows:

The selection of line options available in the History Environment screen is different than the selection of line options available in the Active Environment screen. Below is the Alternate Bottom line of the History Environment screen.

```
Opt: L Log Z Zoom S Stat R Restore J JCL Edit V View Sysout G Group
```

The **OPT** command toggles between Commands and Options display.

Upon exiting the History Environment screen (by pressing **PF03/PF15**), the Active Environment screen is displayed.

#### - NOTE



Group entities are not supported in the History environment. Therefore, jobs originating in group scheduling tables are restored as regular (non-group) jobs.

### **Options of the History Environment Screen**

The History Environment screen has the same options as the Active Environment screen, with the addition of R Restore.

The R Restore option restores the specified job to the Active Jobs file and marks it as deleted in the History Jobs file. The restored job appears in the Active Jobs file in Held status.

For a description of the remaining options, see "Options of the Active Environment Screen" on page 180.

# **Working with different History files**

By default, when the HISTORY command is entered without any parameters, the entries in the default History file are displayed. It's possible to enter the HISTORY command followed by the name of a different history file. As a result, the entries from the "different" history file are displayed.

This feature is available only when using the IOA Online environment under TSO, not under the IOA Online monitor.

### **Force OK Confirmation Window**

To change the status of a job to ENDED OK, type **O** (Force OK) in the option field to the left of the job order and press **Enter**.

Status changes are performed as follows:

- If the job status is WAIT SCHEDULE, the status is changed to ENDED OK without submitting the job. As a result, all resources required by the job are freed, and OUT and SHOUT WHEN OK job post-processing are performed as if the job terminated with status ENDED OK.
- If the job has an ON PGMST code of FORCE, and it terminates with status ENDED NOTOK, the status is changed to ENDED OK. In addition, ON PGMST post processing is performed for the following DO actions: DO COND, DO FORCEJOB, DO SETVAR, and DO SHOUT. For more information about the code FORCE, see the table of parameter values for use with the "ON" Post Processing Parameter in "ON Statements: Post–Processing Parameter" on page 537.

The same effect is achieved if a job has the ON PGMST parameter with value ANYSTEP specified with the code OK, if the FRCOKOPT parameter in the CTMPARM member in the IOA PARM library is set to Y, and if the command FORCE OK is requested.

In the case of a cyclic job, Force OK works only after the job has reached an ENDED status, such as the result of a DO STOPCYCLE command.

A Force OK request is not performed if the job is currently being executed or rerun.

In the case of a Group Entity, Force OK works only if the Group Entity has an ENDED status.

When requesting Force OK, the Force OK confirmation window is displayed, unless the user profile has been modified to suppress the window. The Force OK confirmation window is illustrated in Figure 67.

Figure 67 CONTROL-M Active Environment FORCE OK confirmation window

```
----- CONTROL-M Active Environment
Filter:
COMMAND ===>
                                                                          SCROLL ==> CRSR
O Name
            0wner
                       Odate Jobname JobID
                                                   Тур
                                                                     Status
  DAILYSYS SYSTEM 271207
                                                                 Ĺist
                                   Top of Jobs
                                                   JOB Held Wait Schedule
  DAILYPRD PRODMNGR 301207
DAILYSYS SYSTEM 301207
IOALDNRS PRODMNGR 301207
                                                   JOB Wait Schedule
JOB Wait Schedule
                                                   JOB Wait Schedule
                                                   JOB Wait Schedule
JOB Ended "OK"
JOB Ended "OK"
  IOACLCND PRODMNGR 301207
  TEST3
                       301207
  NOT-SUB
O FAILUNK
                             Force OK
                                                           (Y/N) | eason Unknown Ended-
             Post-processing Options:
OUT Statements Y
                                                                    Forced OK
Forced OK (Run 2)
Ended "OK"
                             ON PGMSTEP Statements Y
SHOUT Statements Y
                                                           (Y/N)
(Y/N)
  WM3714#1
                             CTB STEP Statements
                                                                   (Y/N)
Commands: OPt DIsplay Show HIstory RBal REFresh Auto Jobstat SHPF Note Table
           OPt command toggles between Commands and Options display
```

You can specify in the Force OK confirmation window whether the following DO actions are performed:

- DO COND
- DO FORCEJOB
- DO SET
- DO SHOUT

If you specify that these actions must be performed, this overrides the value specified in the FRCOKOPT parameter in the CTMPARM member in the IOA PARM library. For more information about the FRCOKOPT parameter, see the *INCONTROL for z/OS Installation Guide*.

These actions are only performed if the following additional conditions are satisfied:

- ON PGMST is set to ANYSTEP.
- CODES is set to OK.
- The ON PGMST ANYSTEP statement is not part of a Boolean block.

#### NOTE



If CODES is set to FORCE, the DO actions are performed regardless of the value set in the With Post-Processing field.

Fill in the Force OK confirmation window as follows, and press Enter.

- 1. To confirm the Force OK request, type Y in the Force OK field. To cancel the Force OK request, type N in this field.
- 2. If you do not want the job's OUT condition statements to be performed, type **N** in the OUT Statements field. If you want the job's OUT statements performed, type **Y** in this field.
- 3. If you do not want the DO actions listed above performed, type **N** in the ON PGMSTEP Statements field. If you want these DO actions performed, type **Y** in this field.
- 4. If you do not want the job's SHOUT statements to be performed, type **N** in the SHOUT Statements field. If you want the job's SHOUT statements performed, type **Y** in this field.
- 5. If you do not want the job's CTB STEP statements to be performed, type **N** in the CTB STEP Statements field. If you want the job's CTB STEP statements performed, type **Y** in this field.

#### NOTE



The default values for fields OUT Statements, ON PGMSTEP Statements, SHOUT Statements, and CTB STEP Statements will be taken from the following variables in the user's profile: SACTFOU, SACTFPS, SACTFSH, and SACTFCB (respectively). If the SACTFPS variable is not found in the user's profile, the value set in the FRCOKOPT parameter will be the default for the ON PGMSTEP Statements field. For the other fields, if their respective variables are not found in the user's profile, the default will be Y. The default values will be used if the user profile has been modified to suppress the Force OK Confirmation window.

# **CMEM Rule Definition Facility**

The CONTROL-M Event Manager (CMEM) Rule Definition facility enables you to create, view, or modify CMEM rules for the handling of events in your environment. A CMEM rule consists of parameters that correspond to the decisions and actions to be taken when handling the occurrence of specified external events.

A CMEM rule for a specific situation needs to be defined only once. Once defined, the rule is saved and used as necessary for managing events. CMEM rules may be modified or deleted as required.

CMEM rules are stored in members called rule tables. In many environments, several rules can work together as a group to handle a specific situation. In these cases, it is common to define all such related rules in a single rule table.

Any number of rule tables may be defined, and each rule table may contain any number of CMEM rules.

CMEM rule tables (members) are stored in rule libraries (partitioned data sets). You may define any number of rule libraries.

The number of rule tables in a library, the number of rules in a rule table, and the size of each rule definition, are all calculated dynamically and are not dependent on parameter specifications.

#### NOTE



The CMEM Rule Definition facility does not support members that have been compressed using the ISPF PACK option.

# **Accessing the CMEM Rule Facility**

The CMEM Rule Definition facility contains the following screens:

**Table 86** Rule Definition Facility Screens

Screen	Definition
CMEM Rule entry panel	Allows specification of parameters that determine which screen is displayed.
Table List screen	Displays the list of tables (members) in the specified CMEM rule library.
Rule List screen	Displays the list of rules in the selected table.
Rule Definition screen	Displays the parameters of the selected CMEM rule. This is the main screen of the facility.

To enter the CMEM Rule facility, type **C** in the OPTION field in the IOA Primary Option menu and press **Enter**. The CMEM Rule entry panel is displayed.

# **Creating Tables**

CMEM rule tables can be created in one of the following ways:

■ By typing the new table name in the entry panel and pressing **Enter**. The name of a new rule for the new table may also be entered.

Upon using this method to request that a table be created, a skeletal CMEM rule (that is, one with most fields not filled in) is displayed in the CMEM Rule Definition screen.

Fill in and save this rule definition. The table is created and the rule is the first and only rule in the Rule list of the table. As additional rules are created in the table (described below), they are added to the Rule list.

■ Upon exiting the Rule List screen, if changes have been made to at least one rule, an Exit Option window is displayed. One option of the window allows creation of a new table in which the rules are saved.

# **Creating CMEM Rules**

CMEM rules can be created using the following basic methods:

- A skeletal rule definition can be created by typing the name of a new rule in the entry panel. The table specified in the entry panel may be either a new or an existing table. In this case, the fields in the rule definition are empty.
- A basic copy of an existing rule can be created using the INSERT option in the Rule List screen. In this case, most fields of the new rule definition contain the same values as the fields in the copied rule. The INSERT option is described in "Options of the Rule List Screen" on page 257.

# **Performing Operations on CMEM Tables and Rules**

Many operations can be performed on CMEM rule tables and on the rules contained within them. These operations are performed through commands and options in the various screens of the CMEM Rule Definition facility.

Below is a brief summary of some of the major operations possible in the facility. Options and commands that have not yet been explained are explained in detail following the summary.

#### Accessing (Editing or Browsing) a Table and its Rules

A table (actually, the rules in the table) can be browsed or edited.

When browsed, the table cannot be modified or updated. When the table is edited, new rules may be added and existing rules may be modified or deleted.

Browsing, however, has advantages:

- Access and exit are quicker than in editing.
- A rule list and/or rules that are in use by another user can be viewed.
- Access for browsing might be granted, even though access for editing might be denied due to site security requirements.

To browse a table, and its rule list and the rules it contains, use the BROWSE option in the Table List screen.

Typing the table name in the entry panel or using the SELECT option in the Table List screen provides edit access.

Depending on user profile definitions, if the table requested for editing is in use, either access is granted in Browse mode or access is not granted.

#### **Deleting a Table or a Rule**

Unneeded rules can be deleted by the DELETE option in the Rule List screen, described in "Options of the Rule List Screen" on page 257. Unneeded tables can be deleted by the DELETE option in the Table List screen, described in "Deleting Tables" on page 158.

#### **Ordering Rule Tables**

Rule tables are ordered by option ORDER or FORCE in the Table List screen, discussed in "Ordering CMEM Rule Tables" on page 267.

#### **Saving Modifications**

All changes made to a table and its rules are kept in memory until the table is exited. Upon exiting the table, you may choose to save or cancel the changes, as described in "Exiting the CMEM Rule Definition Facility" on page 264.

# **Entry Panel**

The entry panel is displayed upon entering the CMEM Rule Definition facility (option C on the IOA Primary Option menu).

Figure 68 CMEM Rule Definition Facility – Entry Panel

```
COMMAND ===>

SPECIFY LIBRARY, TABLE NAME, RULE NAME

LIBRARY ===> CTM.PROD.RULES

TABLE ===> (Blank for table selection list)

RULE ===> (Blank for rule selection list)

USE THE COMMAND SHPF TO SEE PFK ASSIGNMENT 22.35.51
```

### **Fields of the Entry Panel**

Fill in the following fields and press Enter.

**Table 87** Fields of the Entry Panel

Field	Description
LIBRARY	Name of the desired CMEM rule library. Mandatory.
	If this field is specified without filling in the TABLE field, the list of tables in the specified library is displayed in the Table List screen.
TABLE	Name of the desired rule table. Optional.
	If this field is specified without filling in the RULE field, the list of rules in the specified member is displayed in the Rule List screen. If a new table name is specified, a new rule definition is displayed in the Rule Definition screen.
RULE	Name of the desired rule. Optional.
	This field can be specified only if a TABLE value is entered. If specified, the requested rule is displayed in the Rule Definition screen.



#### – NOTI

If you use the selection list fields, their values are not erased until you exit the entry panel by pressing END (PF03/PF15).

### **Table List Screen**

The Table List screen displays a list of rule tables (members) in the specified library. This screen can be entered directly from the entry panel or upon exiting the Rule List screen.

By default, only table names are listed in the screen. However, if the default has been modified at time of installation, statistical information is displayed for each table name (as shown below).

Figure 69 CMEM Definition Facility Table List Screen

TABLES OF LIBE	RARY CTM.PROD.I	RULES				-			( - /
COMMAND ===>	\/\/_MM	CDEATED	CHANC	- F.D.	CIZE	TNITT	SCROLI		CRSR
OPT NAME PRDJACCT	VV.MM		01/06/06	14.20		INIT 56	MOD	ID M06	
PRDJPYRL			01/06/06					M86B	
PRDJFNC			01/06/06		6	6		NO4A	
PRDJMRKT			01/06/06					NO4A	
BACKUP			01/06/06		61			M06	
TESTJ			01/06/06		6	56		M06	
	01.00 NO <>>>>								
OPTIONS: S SI	ELECT O ORDE	R F FOR	CE B BRO	WSE I	D DELET	ГЕ		12.1	1.50

To return to the entry panel, press the END key (PF03/PF15).

### **Options of the Table List Screen**

To request one of the following options, type the option in the OPT field to the left of the table names and press **Enter**.

Table 88 Options of the Table List Screen

Option	Description
S (SELECT)	Display the list of rules in the table for any purpose, including editing or modification. Only one table can be selected at a time.
O (ORDER)	Order all rules in the table, as described in "Ordering CMEM Rule Tables" on page 267. Multiple tables can be ordered.
B (BROWSE)	Display a list of rules in the table for browsing. Only one table at a time can be browsed.
F (FORCE)	Order all rules in the table. Because CMEM rules have no basic scheduling parameters, this option works like the Order option. Multiple tables can be forced.
D (DELETE)	Delete the table (member) from the library. This is discussed in "Deleting Tables" on page 158. Multiple tables can be deleted.

#### NOTE



Users whose access to options has been limited by the INCONTROL administrator can only access the Browse option.

### **Rule List Screen**

The Rule List screen displays the list of CMEM rules in the specified CMEM table. The following fields are listed for each rule:

**Table 89** Fields of the Rule List Screen

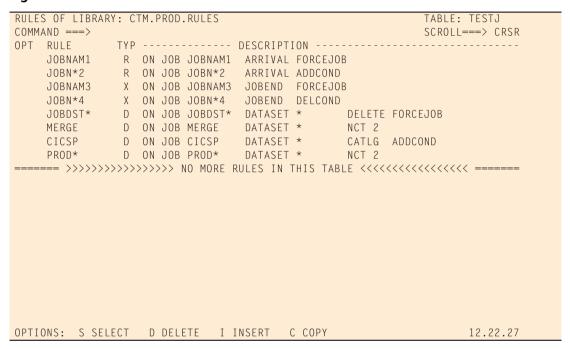
Field	Description
OPT	Select, Delete and Insert options can be entered in this field, as described in "Options of the Rule List Screen" on page 257.
ТҮРЕ	The event type of the rule. The following event type codes exist:  ■ R – Job arrival ■ X – Job end ■ D – Data set ■ Z – Step
DESCRIPTION	The description of the rule. This appears in the DESCRIPTION field of the rule definition.

This screen can be entered directly from the entry panel or the Table List screen, or upon exiting from the Rule Definition screen.

#### - NOTE -

If the S (Select) option was typed in the Table List screen for a table that is currently in use ("selected") by another user, either the Rule List screen is not displayed and the Table List screen remains displayed (the default), or the Rule List screen is displayed in Browse mode (if a user profile definition overrides the default). In either case, an appropriate message is displayed.

Figure 70 CMEM Rule Definition Rule List Screen



### **Commands of the Rule List Screen**

The following commands may be typed in the **COMMAND** field of the Rule List screen.

Table 90 Commands of the Rule List Screen (part 1 of 2)

Command	Description
	The DESC command displays the rule description next to the rule name. The description is taken from the DESCRIPTION field in the rule.

Table 90 Commands of the Rule List Screen (part 2 of 2)

Command	Description
STAT	The STAT command displays the following ISPF-like statistical information about the rule next to the rule name: version and modification numbers, creation date, last modification date, and user ID.
SORT	Sorts the list of rules in the Rule List screen according to specified criteria. Valid values are:  R (rule) - Sorted according to rule name T (type) - Sorted according to rule type

### **Options of the Rule List Screen**

To request one of the following options, type the option in the OPT field to the left of the rule names and press **Enter**.

#### _ NOTE



If the Rule List screen is displayed in Browse mode, options D (Delete) and I (Insert) are not available.

Table 91 Options of the Rule List Screen

Option	Description
S (SELECT)	Display the Rule Definition screen with details of the specific rule. Only one rule can be selected at a time.
	If the Rule List screen is not displayed in Browse mode, the rule definition may be edited and updated. If the Rule Definition screen is displayed in Browse mode, the rule definition may only be browsed; it cannot be modified.
D (DELETE)	Delete a rule from the Rule list (member). Multiple rules can be selected for deletion.
I (INSERT)	Insert a new rule in the list. The Rule Definition screen appears, with the same details of the rule marked "I". Only one rule may be inserted at a time.
C (COPY)	Copy the rule to another table. Multiple rules can be selected. For more information, see "Copying Rules to Another Table" on page 269.

## **Rule Definition Screen – Defining Rules**

The Rule Definition screen is used to define, display and modify parameters of a specific CMEM rule.

This screen can be entered directly from the entry panel or from the Rule List screen. Update of parameters is not permitted in Browse mode.

The rule definition may take up more than one screen.

To delete a parameter on the screen, simply erase it by the EOF key or blank it out. If additional action is required, CONTROL-M issues appropriate instructions.

Figure 71 Rule Definition Screen - Defining Rules

```
RL: BKP*
            LIB CMEM.PROD.RULES
                                                      TABLE: BACKUP
COMMAND ===>
                                                      SCROLL===> CRSR
  ON JOBARRIV = BKP* JTYPE SMFID SYSTEM
                                                       And/Or/Not
  OWNER ADMIN GROUP BACKUP
                                       MODE PROD RUNTSEC
  THRESHOLD
  DESCRIPTION MONITOR STARTUP OF BACKUP JOBS
  DESCRIPTION
  /* TELL CONTROL-M TO MONITOR THIS JOB
  DO FORCEJOB = TABLE BACKUP JOB
                                                DATE ODAT
            LIBRARY CTM.PROD.SCHEDULE
  D0
----- >>>>>>>> END OF RULE DEFINITION PARAMETERS <<<<<<<<<<
FILL IN RULE DEFINITION. CMDS: CAPS, EDIT, SHPF
                                                            21.00.36
```

Rule parameters are divided into the following basic groups:

- Event Selection parameters
- General parameters
- Action parameters

A brief explanation of available parameters follows. For a detailed explanation of each rule parameter, see Chapter 4, "CONTROL-M Event Manager (CMEM)."



#### - NOTE -

Parameters marked with the  M  symbol may have many occurrences. Whenever you fill in the last occurrence of the parameter on the screen, CONTROL-M adds a new empty occurrence of the parameter that can be filled in. The only limit to the number of occurrences is the region size available for the application.

### **Event Selection Parameters**

Event Selection parameters specify selection criteria under which actions are performed by CMEM.

Figure 72 CMEM Rule Definition Event Selection Parameters - Example

ON JOBARRIV = A* JTYPE J SMFID ON JOBARRIV = C* JTYPE J OWNER ADMIN GROUP CICS	SYSTEM And/Or/Not O And/Or/Not MODE RUNTSEC
ON JOBEND = CICSPROD JTYPE SMFID OWNER ADMIN GROUP CICS	SYSTEM And/Or/Not MODE PROD RUNTSEC
ON DSNEVENT = PRD0001 JTYPE SMFID  DSN PROD.*  PROCSTEP PGMSTEP	SYSTEM  DISP NCT2  STEPRC And/Or/Not
OWNER ADMIN GROUP PRODJOBS	MODE RUNTSEC
ON STEP = PRDOOO1 JTYPE SMFID PROCSTEP STEP2 PGMSTEP OWNER CTMCTLM GROUP	SYSTEM STEPRC OK And/Or/Not MODE PROD RUNTSEC NONE

**Table 92 CMEM Rule Definition Event Selection Parameter** 

Parameter	Description
ON statementM	Conditions under which the rule is to be performed. Subparameters may be displayed.
	Valid ON statement values are:
	<ul> <li>ON JOBARRIV – Job name (or mask) of a job or started task that arrived on the JES spool from any source.</li> </ul>
	<ul> <li>ON JOBEND – Job name (or mask) of a job or started task that terminated.</li> </ul>
	<ul> <li>ON DSNEVENT – Name (or mask) of a job or started task or TSO user to be monitored for data set events (including NOT CATLGD 2 events).</li> </ul>
	<ul> <li>ON STEP – Name (or mask) of a job procedure step (and optionally, program step) that terminated, and its desired return code.</li> </ul>
AND/OR/NOTM	Conjunctional subparameter that enables linking of ON statements.

### **General Parameters**

The following are General parameters that apply to the rule.

### Figure 73 General Parameters

ON JOBARRIV =	= BKP*	JTYPE	SMFID	SYSTEM		And/Or/Not
OWNER ADMIN	GROUP	BACKUP		MODE	PROD	RUNTSEC
THRESHOLD						
DESCRIPTION N	MONITOR	STARTUP OF	BACKUP	JOBS		
DESCRIPTION						

#### **Table 93 CMEM Rule Definition General Parameters**

Parameter	Description
OWNER	ID of user requesting CMEM services.
GROUP	Logical name of a group of rules.
MODE	CMEM rule operation mode.
RUNTSEC	Type of runtime security checks to be performed by the rule.
DESCRIPTION	Free text description of the rule definition that appears in the Rule List screen.

### **Action Parameters**

Action parameters specify actions to be performed by CMEM.

**Table 94 CMEM Rule Definition Action Parameters** 

Actions to be performed when the rule is triggered. They are
performed sequentially. Valid DO statements are illustrated below:

DO COND	Add or delete a prerequisite condition.
	p q

DO FORCEJOB = TABLE	BACKUP	JOB	DATE ODAT
LIBRARY CTM.PROD.SCHE	DULE		
DO			

DO FORCEJOB	Force one or more jobs under CONTROL-M.

DO STOPJOB DO

DO STOPJOB	Stop execution of the remaining steps of the job that triggered the
	rule.

The following DO statements are available in CMEM only if CONTROL-O is active. For more information on the subparameters of the DO parameter which appear in these illustrations, see the *CONTROL-O User Guide*.

DO SHOUT = TO TSO-DBA	URGENCY U SYSTEM
MESSAGE DB2 MASTER ENDED -	PLEASE CHECK!
DO	

DO SHOUT	Issue a message to a console, TSO user, ROSCOE user, IOA Log or
	the system administrator using the CONTROL-O Shout facility.

DO RULE	= PROCFILE %%\$DSN	OWNER PROD	
TABLE	PRODRULE LIBRARY CTO.PROD.RULES		
SYSTEM	SHARELOC NO TIMEOUT NO		
DO DO			

DO RULE	Invoke a CONTROL-O rule from within the current rule.

### **Commands of the Rule Definition Screen**

The following commands may be typed in the COMMAND field of the Rule Definition screen.

Table 95 Commands of the Rule Definition Screen

Command	Description
CAPS	By default, all entries of lowercase characters are converted and saved as uppercase. In CMEM rules, certain fields, such as the string entered in the ON SHOUT MS field, can be enabled to accept and save lowercase characters, by using the CAPS OFF command, as described below. Valid formats are:
	<ul> <li>CAPS OFF – Enables certain user entries to be saved and displayed in lowercase characters.</li> </ul>
	<ul> <li>CAPS ON – Forces all user entries to be displayed in uppercase characters, regardless of the case in which they were entered. Default.</li> </ul>
	<ul> <li>CAPS – Indicates whether CAPS ON or CAPS OFF mode is active.</li> </ul>
	<b>Note:</b> JCL jobs used by CONTROL-M do not support lowercase characters. Using lowercase characters to define IOA variables is not recommended if those variables are shared by CONTROL-M through IOAVAR.
EDIT	Alternately enters and exits the Edit environment of the Rule Definition screen. The Edit environment provides ISPF-like line editing commands to the Rule Definition screen. For more information, see Appendix D, "Editing CMEM Rule Definitions in the Edit Environment."
SHPF	Shows the current PFKey assignments.

Commands used to exit the Rule Definition screen are described in "Exiting the Rule Definition Screen" on page 264.

## **Entering Comments**

Comments are free text descriptions of rule definition parameters that are stored in a rule definition. It is recommended that comments be inserted within rule definitions for clarification and documentation purposes. Comment lines begin with the symbols /*, and are not processed during rule execution.

Use one of the following methods to insert comment lines:

■ Decide where you want the comment to be inserted. Position the cursor on the preceding line, and press **CMNT** (**PF04/PF16**) to open the comment line. If you need more than one line, press **Enter** at the end of each line.

- Type CMNT in the COMMAND field and move the cursor to the line before which the comment is to be inserted. Press Enter.
- To insert comments between DO statements an additional method is available. Type /* in an empty DO statement and press **Enter**.

Comment usage is illustrated in the following Rule Definition screen:

Figure 74 Rule Definition Screen Comment Usage

```
LIB CMEM.PROD.RULES
RI: BKP*
                                          TABLE: BACKUP
COMMAND ===>
                                          SCROLL===> CRSR
+-----+
 OWNER ADMIN GROUP BACKUP
                              MODE PROD RUNTSEC
 THRESHOLD
 DESCRIPTION MONITOR STARTUP OF BACKUP JOBS
 DESCRIPTION
 /* TELL CONTROL-M TO MONITOR THIS JOB
 /*
 DO FORCEJOB = TABLE BACKUP JOB
                                      DATE ODAT
          LIBRARY CTM.PROD.SCHEDULE
 D0
 ---- >>>>>>>> END OF RULE DEFINITION PARAMETERS <<<<<<<<<<<
FILL IN RULE DEFINITION. CMDS: CAPS, EDIT, SHPF,
```

An unlimited number of comment lines may be entered within a rule definition.

### **Editing CMEM Rule Definitions in the Edit Environment**

Rule Definition parameters can be edited (moved, copied, deleted, repeated) using CMEM Line Editing commands, similar to standard ISPF line commands, from within the CMEM Edit environment.

The Edit environment in a Rule Definition screen is accessed by typing EDIT in the **COMMAND** field and pressing **Enter**.

Details and examples of the editing CMEM rule definitions in the Edit environment are provided in Appendix D, "Editing CMEM Rule Definitions in the Edit Environment."

## **Exiting the CMEM Rule Definition Facility**

When exiting the CMEM Rule Definition facility, screens are exited in the following sequence:

- Rule Definition screen
- Rule List screen
- Table List screen

#### NOTE



If the Table List screen was bypassed when you entered the CMEM Rule Definition facility (that is, if you entered a TABLE value in the entry panel), the Table List screen is not displayed upon exiting the Rule List screen; instead, the entry panel is displayed.

■ Entry panel

The commands and options available when exiting screens depend on the screen being exited and on whether changes have been made. If changes have been made, the selected exit options and commands determine whether the changes are saved. Exit options and commands are discussed below on a screen-by-screen basis.

### **Exiting the Rule Definition Screen**

Use any of the following commands, or press the corresponding PFKey, to exit the Rule Definition screen:

Table 96 Commands for Exiting the Rule Definition Screen

Command	Description		
CANCEL	Cancel the changes made to the rule and return to the Rule List screen.		
END ( <b>PF03/PF15</b> )	Keep changes to the rule in memory and exit to the Rule List screen.		

#### NOTE



The END command retains changes to the rule in memory. To permanently save the changes to disk, you must request that the changes be saved when you exit the Rule List screen.

### **Exiting the Rule List Screen**

Use the END command (**PF03/PF15**) to exit the Rule List screen. If changes made to at least one rule have been kept in memory (as described in the preceding section) and/or if any changes have been made to the Rule List screen, the Exit Option window is displayed.

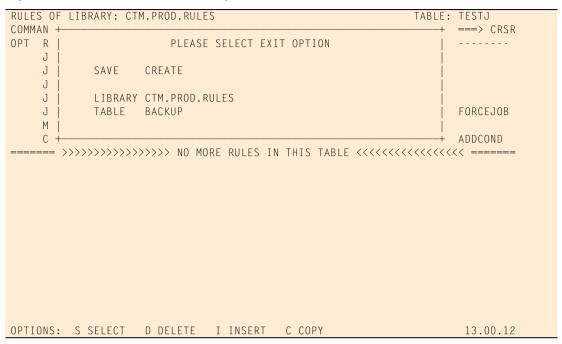


Figure 75 Rule List Screen Exit Option Window

Fill in the Exit Option window as follows:

The LIBRARY and TABLE fields indicate the library and table in which the rule definitions are saved. The specified values can be modified (for example, to save the rule definitions in a new or different table).

- To save all changes currently in memory and exit the Rule List screen, type Y (Yes) after the word SAVE or CREATE:
  - Type Y after SAVE if a table with the same name already exists in the specified library.
  - Type Y after CREATE if a table with the same name does not exist in the specified library.





If you create a new table, the table name does not appear in the Table List screen upon exiting the Rule List screen; it first appears when you reenter the Table List screen from the entry panel.

- To cancel changes currently in memory and exit the Rule List screen, type N (No) after the word SAVE or CREATE.
- To close the Exit Option window and remain in the Rule List screen (with the changes remaining in memory), press the RESET key (PF04/PF16).

### **Exiting the Table List Screen**

Press the END key (PF03/PF15) to exit the Table List screen.

### **Exiting the Entry Panel**

Press the END key (PF03/PF15) to exit the entry panel.

### **Deleting Tables**

Tables can be deleted from the Table List screen.

To delete tables, type option D (Delete) by the table names in the Table List screen and press **Enter**.

The confirmation window illustrated below is displayed, in sequence, for each table selected for deletion.

Figure 76 Rule Definition Facility Delete Table Confirmation Window

```
TABLES OF LIBRARY CTMP.V600TST.RULES
COMMAND ===>
                                                SCROLL===> CRSR
OPT NAME ----- VV.MM CREATED
                                CHANGED SIZE INIT MOD ID
           0 M06
    PRDJACCT
                                                    0 M86B
    PRDJPYRL
                                               6
                                                    0 NO4A
    PRDJFNC
                        CUNFIRM DELETE OPTION | 5
(Y/N) | 61
                       CONFIRM DELETE OPTION |
    PRDJMRKT
    TESTJ
----- >>>>>>>> NO MORE TABLES IN THIS LIBRARY <<<<<<<<<
OPTIONS: S SELECT O ORDER F FORCE
                               B BROWSE
                                       D DELETE
                                                      13.05.52
```

Type **Y** (Yes) in the window to confirm the delete request.

Type N (No) in the window to cancel the delete request.

A message is written to the IOA Log file for each table deleted.

#### — NOTE



If PDSMAN is operational at your site, \$\$\$PACE members are not deleted.

### **Ordering CMEM Rule Tables**

A rule definition that resides in a library is not active until it has been ordered. Rule tables can be ordered automatically when CMEM is started, as described in the *INCONTROL for z/OS Administrator Guide*, or they can be ordered manually. Regardless of the method used to order them, CMEM rule tables only need to be ordered once. Once a rule table is ordered, it remains active unless replaced or deleted by an operator command, or until CMEM is stopped.

When rule definitions are updated or modified, the rule table must be ordered again. The newly ordered version of the rule table automatically replaces the previous version of the rule table. Rule tables can be deleted from the Active environment by an operator command. For details, see Chapter 6, "Selected Implementation Issues."

To order a rule table, type O (Order) or F (Force) in the OPT field to the left of the table name in the Table List screen. Because there are no basic scheduling criteria in CMEM rules, the Order and Force options work the same way. More than one rule table can be ordered at the same time.

When you order rule tables, the following default confirmation window is opened. If the default has been modified in the user profile, a double confirmation window is opened. For more information on the double confirmation window, see "The Double Confirmation Window" on page 155.

Figure 77 Order and Force Confirmation Window

	ES OF LIBRARY CTM.P	ROD.RULES			(C) SCROII===> CRSR
0 O	PRDJACCT PRDJPYRL PRDJFNC PRDJMRKT <	01.06 01/02/14 +   CONFIRM   ASK FOR	01/06/06 14:29  DDATE 06060	56 01    +	56 0 M86B 6 0 N04A 5 0 N04B 56 0 M06
	>>>>>>>>>>>>>				
OPTI	ONS: S SELECT O	ORDER F FORCE	B BROWSE D D	ELETE	12.11.50

**Table 97 Options for Ordering Rule Tables** 

Option	Description					
CONFIRM	Whether to process the order request. Valid values are:  ■ Y (Yes) – Process the request ■ N (No) – Cancel the request					
ODATE	Current date (in mmddyy, ddmmyy or yymmdd format, depending on the site standard).					
ASK FOR EACH ONE	This line is displayed only if more than one table order is requested. It determines whether individual confirmation is required for each order request. Valid values are:					
	■ Y (Yes) – Individual confirmation is required for each order request. The selected CONFIRM value (Y or N) applies only to the current order or request.					
	■ N (No) – Individual confirmation is not required for each order request. The selected CONFIRM operation is applied to all order requests. If CONFIRM is Y, all order requests are processed; if CONFIRM is N, no order requests are processed.					

When you press **Enter**, the results of the order request are displayed in a message at the top of the screen. If more than one message is required, the original list screen disappears and the messages appear in a new screen. If the messages span more than one screen, you may scroll up and down on the messages list. Press the **END** key **(PF03/PF15)** to return to the Table List screen.

### **Wish W00945**

As a result of the Order or Force request an MVS MODIFY command is sent to the CMEM monitor. Some installations may protect the MVS command. If the user is not authorized to issue an MVS MODIFY command the Order or Force fails.

After applying this wish, by specifying WO0943 APPLY=YES in the IOADFLTL member, the MODIFY command will be executed under the CMEM monitor. Therefore, authority to issue a MVS MODIFY command is required only for the CMEM or CMEM monitor USERID.

#### Wish WO0945 also introduces

- the following AutoEdit System variables that return Wish WO0943 with a status of Y or N
  - %%\$WO0943
  - %%\$MODIFY_O_F
- the following CMEM MODIFY command

F controlo, WISH=WO0943=xxxxx

where xxxxxx - ENABLE or DISABLE

This command allows the user to enable or disable optional Wish WO0943 without stopping CONTROL-O.

#### - NOTE

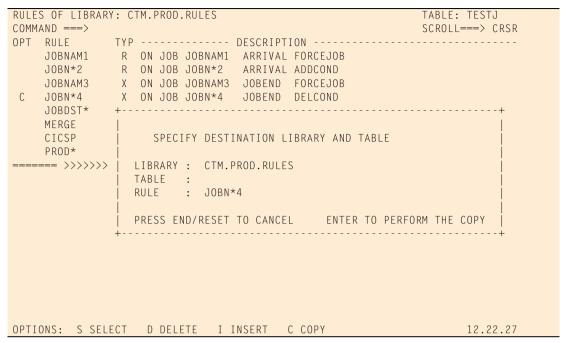


This is a temporary change to the Wish only. In order to make the change permanent, Wish WO0943 must set to the required value, YES or NO, in member IOADFLTL.

### **Copying Rules to Another Table**

To copy one or more rules from the current table to another table, type C (Copy) in the OPT column of the Rule List screen, next to the rule names, and press **Enter**. The following window is displayed:

Figure 78 Window for Copying Rules to Another Table



The window contains the fields shown in the following table. Some fields contain default values that can be modified.

**Table 98** Fields in the Window for Copying Rules to Another Table

Field	Description				
LIBRARY	Library containing the table into which the rules must be copied.  Must be an existing library. Default is the current library.				
TABLE	Name of the table into which the rule must be copied.				
	<b>Notes:</b> A rule can only be copied to another table. It cannot be copied to its own table (even if the rule is renamed).				
	If the selected table does not exist, the table is created when the request is performed.				
RULE	Name of the rule to be copied. If multiple rules are selected, the window initially display with the first selected rule. As each request is performed or canceled, the next requested rule name is displayed.				

To perform a request, press **Enter**.

To cancel a request, press END (PF03/PF15) or RESET (PF04/PF16).

## **IOA Variables Database Facility**

The IOA Variable Database facility is composed of a series of screens that enable you to view, create, and modify Global AutoEdit variables in IOAVAR, the IOA Global Variable database.

As of version 6.0.00, this facility is available to CONTROL-M and/or CONTROL-O users only, and is available only if CMEM and/or CONTROL-O are installed. CONTROL-O users can also create and modify other variable databases.

#### NOTE



If CONTROL-O is installed, the CMEM facility is controlled by the CONTROL-O monitor instead of the CMEM monitor. In this case, references to the CMEM monitor apply instead to the CONTROL-O monitor, and operator commands must contain **CONTROLO** in place of **CTMCMEM**.

Certain screens and features of the IOA Variable Database Facility are relevant to CONTROL-O but not to CONTROL-M. If CONTROL-O is installed, and you are using the IOA Variable Database Facility with CONTROL-O, refer to the description of the IOA Variable Database facility in the CONTROL-O User Guide.

Global variables may be defined in variable database IOAVAR. They can be accessed and updated by all CONTROL-M jobs and/or CONTROL-O rules on that computer. In addition, if Sysplex support is installed, these variables can be accessed and updated by all CONTROL-M jobs and CONTROL-O rules across the Sysplex. IOA Global variable database administration and Sysplex support are described in detail in the *INCONTROL for z/OS Administrator Guide*.

#### NOTE -



In addition to using the online IOA Variable Definition facility to define Global variables, Global variables can be defined or updated through SET VAR and DO SET statements in CONTROL-M job scheduling definitions, through SET statements in the job's JCL, and through SETOGLB statements in a KSL or KOA script. For more information on KeyStroke Language (KSL) User Guide, refer to the *KeyStroke Language (KSL) User Guide*.

The IOA Variable Database facility consists of several screens, some of which are relevant only to the INCONTROL administrator, and some of which are relevant to all users. The following screens are relevant to all users. Screens relevant only to administrators are described in the *INCONTROL* for *z/OS* Administrator Guide.

Table 99 IOA Variable Database Facility Screens (part 1 of 2)

Screen	Description
Variable Database entry panel	Enables specification of a database name. If no database name is entered, the Database List screen is displayed.
Database List screen	Displays the list of variable databases.

Table 99 IOA Variable Database Facility Screens (part 2 of 2)

Screen	Description
Values of Database screen	Displays the list of variables, and their values, in the selected database.
Variable Zoom screen	Displays the complete variable name and the complete variable value of the selected variable, and enables update of the variable value.

To enter the Variable Database facility, select option IV in the IOA Primary Option menu. The IOA Variables Entry Panel is displayed.

### **Entry Panel**

The following entry panel is displayed upon specification of option IV in the IOA Primary Option menu:

Figure 79 IOA Variable Database Entry Panel

```
COMMAND ===>

SPECIFY DATABASE NAME

DATABASE ===>

(Blank for Database selection list)

MODE ===>

(Blank or ADMIN for Administration mode)

USE THE COMMAND "SHPF" TO SEE PFK ASSIGNMENT

16.21.43
```

- To display the list of all variable databases: Press Enter. The Database List screen is displayed.
- To display the list of variables in a specific variable database: Type the database name in the DATABASE field and press Enter. The Values of Database screen for the specified database is displayed. An asterisk (*) can be used as a mask character in the DATABASE field of this screen.

#### – NOTE



CONTROL-O can have multiple variable databases; CONTROL-M can only use the IOAVAR database.

- To enter the IOA Variables Database facility as a regular user: Leave the MODE field blank. In this case, you will be able to view the variables in the databases, but you will not be able to perform administration functions.
- To enter the IOA Variables Database facility with full administrator functionality: Type ADMIN in the MODE field.

#### NOTE -



The additional functions available in administration mode are described in the *INCONTROL for z/OS Administrator Guide*.

### **Database List Screen**

The Database List screen (also called the IOA Variables In Use screen) displays a list of the variable databases currently defined. This screen can be entered directly from the entry panel or when returning from the Variable List screen.

#### NOTE



For a CONTROL-O variable database to be loaded into memory, the database must be listed in the DAGLBLST DD statement of the CMEM monitor. For more information, see the description of the IOA Variable Database facility in the *INCONTROL for z/OS Administrator Guide*.

Figure 80 IOA Variable Database Facility Database List Screen

```
IOA VARIABLES IN USE
COMMAND ===>
                                                          SCROLL ===> CRSR
OPT NAME
                       DESCRIPTION
                       COSMOS - DEMO - METHODS DATABASE
   COSALLMT
   COSALLPR
                       COSMOS - DEMO - PREREQUISITES DATABASE
   COSIMGOB
                       COSMOS - DEMO - SYSIMAGE WORKING DATABASE
   IOAVAR
                       IOA GLOBAL VARIABLE DATABASE
                       COSMOS - PROD - METHODS DATABASE
   PRDALIMT
   PRDALLPR
                       COSMOS - PROD - PREREQUISITES DATABASE
   PRDSTCOB
                       COSMOS - PROD - STC WORKING DATABASE
                       COSMOS - PROD - STC SOURCE DATABASE
   PRDSTCSD
                       COSMOS - PROD - VTAM WORKING DATABASE
   PRDVTMOB
                       COSMOS - PROD - VTAM SOURCE DATABASE
   PRDVTMSD
                       AUTOEDIT DATABASES - TUTORIAL
   TUTORIAL
                       XESXAE - DEMO - S1 - TEMP
   XAES1D01
                       XESXAE - DEMO - S1 - INPUT
   XAES1D02
                       XESXAE - DEMO - S1 - PROT (S1PROT SAME AS INPUT)
   XAES1D03
                       XESXAE - DEMO - S1 - INOUT (S1INOUT SAME AS INPUT)
   XAFS1D04
   XAES2D01
                       XESXAE - DEMO - S2 - TEMP
   XAES2D02
                       XESXAE - DEMO - S2 - INPUT
   XAES2D03
                       XESXAE - DEMO - S2 - PROT (S2PROT SAME AS INPUT)
   XAES2D04
                       XESXAE - DEMO - S2 - INOUT
  OPTIONS: V VIEW VARS
                                                               08.15.55u
```

A short description is displayed for each database. You should create a description when creating a new database.

### **Options of the Database List Screen**

To request an option, type the option in the OPT field to the left of the database name, and press **Enter**. The V (VIEW VARS) option is available for all users. This option displays the variables of the database in the Values of Database screen.

#### NOTE -



Only option V is intended for all users. The remaining options (I - Insert, U - Update, and S - Select) are intended for INCONTROL administrators only, and are described in the *INCONTROL* for *z/OS* Administrator Guide.

### **Values of Database Screen**

The Values of Database screen can be entered directly from the entry panel or from the Database List screen.

Each line in the screen represents a variable in the database. Variables and their values are loaded into memory automatically at CONTROL-M startup.

Figure 81 IOA Values of Database Screen

VALUES OF DATA	BASE: IOAVAR	(IV.V)
COMMAND ===>		SCROLL===> CRSR
ROW VARN	IAME	VALUE
00022889 %%\M\	ACCTS\BCKP\PDTAPE_0001_1	045673
00023866 %%\M\	ACCTS\BCKP\PDTAPE_0001_2	1045683
00024902 %%\M\	ACCTS\BCKP\PDTAPE_0001_3	045677
00025863 %%\M\	ACCTS\BCKP\PDTAPE_0001_4	043433
00026943 %%\M\	ACCTS\BCKP\PDTAPE_0001_5	045543
00027792 %%\M\	ACCTS\BCKP\PDTAPE_0001_6	045556
00028972 %%\M\	ACCTS\BCKP\PDTAPE_0001_7	045666
00029831 %%\M\	ACCTS\EMPLY\EMP_00123_SCH00L	STATE UNIVERSITY OF NEW YORK A
00030765 %%\M\	OPER\KPL\SPACE_TYPE_5	TRK
00031985 %%\M\	SYS\DBLG\NAME_OF_COMPUTER_1	A
00032972 %%\M\	SYS\DBLG\NAME_OF_COMPUTER_2	D
00033769 %%\M\	SYS\DBLG\NAME_OF_COMPUTER_3	K
00034919 %%\M\	SYS\DBLG\NAME_OF_COMPUTER_4	W
00035955 %%\M\	OPER\GRPBKP\GENERATION NUMBER A	001
00036932 %%\M\	OPER\GRPBKP\GENERATION NUMBER B	001
00037778 %%\M\	OPER\GRPBKP\GENERATION NUMBER C	003
00038808 %%\M\	OPER\GRPBKP\GENERATION NUMBER D	002
	>>>>>>> NO MORE ROWS FOR THIS	
OPTIONS: Z ZO	OOM I INSERT D DELETE R RE	PEAT 09.19.41

The Values of Database screen displays the following information about the variables in the IOAVAR database:  $\frac{1}{2} \frac{1}{2} \frac{1}{2$ 

Table 100 Fields of the IOA Values of Database Screen (part 1 of 2)

Field	Description
ROW	Each variable is assigned its own row in the database. This column displays the row number of the variable.

Table 100 Fields of the IOA Values of Database Screen (part 2 of 2)

Field	Description			
VARNAME	The variable path and name, with the following format:			
	%%\M\app_name\group_name\job_name\var_name			
	where:			
	<ul> <li>%% – Indicates that the string is a variable. Constant.</li> <li>M – Indicates that the string is a CONTROL-M variable. Constant. Mandatory.</li> <li>app_name – The CONTROL-M application where var_name resides. Optional.</li> <li>group_name – The CONTROL-M group within app_name where var_name resides. Optional.</li> <li>job_name – The CONTROL-M job within group_name where var_name resides. Optional.</li> <li>var_name – The variable name. Mandatory.</li> </ul>			
	Up to 30 characters of the VARNAME string are displayed. If the VARNAME string is longer, the full variable path and name can be viewed in the Variable Zoom screen, which is described in "Variable Zoom Screen" on page 278.			
	<b>Note:</b> All levels in the path within the VARNAME string are separated by a $\setminus$ (backslash).			
VALUE	Value of the variable. Up to thirty characters of the value are displayed. If the value is longer, the full value can be viewed in the Variable Zoom screen, which is described in "Variable Zoom Screen" on page 278.			

Use the scrolling PFKeys to scroll the variable database LEFT (**PF10/PF22**) and RIGHT (**PF11/PF23**).

### **Options of the Values of Database Screen**

To request one of the following options, type the option in the first column of the row number to the left of the variable name and press **Enter**.

Table 101 Options of the Values of Database Screen (part 1 of 2)

Option	Description
Z (ZOOM)	Display the full variable and value of the selected row (variable) in the Variable Zoom screen. The displayed variables may be edited in this screen, as well. For more information, see "Variable Zoom Screen" on page 278.
I (INSERT)	Insert a new row in the variable database. For more information, see "Adding a Row (Variable)" on page 277.

Table 101 Options of the Values of Database Screen (part 2 of 2)

Option	Description
D (DELETE)	Delete the selected row (variable).
R (REPEAT)	Insert a new row that is identical to the one for which this option is selected. For more information, see "Adding a Row (Variable)" on page 277

### Adding a Row (Variable)

A row (variable) can be added to the database using either option I or option R:

■ Option I (Insert)

This option is useful for defining a variable that is not similar to the one it follows. When option I is typed next to a variable, a row is immediately inserted below the selected row, and a row value is assigned, as explained in "Row Numbering" on page 277. However, the VARNAME and VALUE fields are blank.

Option R (Repeat)

This option is useful for defining a variable that is similar to the one it follows. When option R is typed next to a variable, a row is immediately inserted below the selected row, and a row value is assigned, as explained in "Row Numbering" on page 277. The new row contains the same VARNAME and VALUE as the selected row.

Using either method, the new row must be edited:

- If Option I was entered, a VARNAME and VALUE must be added to the new row.
- If Option R was entered, the repeated VARNAME must be changed because each variable in the database must be unique. The VALUE may also be changed if desired.

The VARNAME and VALUE in the new row can only be edited in the Variable Zoom screen, described below.

### **Row Numbering**

Row numbers in a variable database are initially incremented by 1000.

- When a new row is inserted (by option I or R), it is assigned an intermediate number incremented by 100.
- Rows inserted between row numbers with a hundreds value are assigned numbers incremented by ten.

■ Rows inserted between row numbers with a tens value are assigned numbers incremented by one.

For example, a row inserted immediately after row 2000 is assigned a number of 2100.

A maximum of 999 rows may be inserted between two original rows in a variable database.

Row numbers can be refreshed (that is, assigned new numbers incremented by 1000) in the following way:

- 1. Unload the IOA Variable Database Variables file using job IOAVARUL in the IOA JCL library. This job invokes the IOADUL utility.
- 2. Reload the file using job IOAVARLD in the IOA JCL library. This job runs the IOADLD utility with the RENUM parameter specified.

For more information about the IOADUL and IOADLD utilities, see the *INCONTROL* for *z/OS Utilities Guide*.

### Variable Zoom Screen

The Variable Zoom screen is used to display the full variable name and path, and the full variable value. The screen is also used to update the variable name and path, and its assigned value.

The full name and path of each variable, and the value of each variable, can be up to 140 characters in length. However, only thirty characters each of the variable and its value can be displayed in the Values of Database screen.

The Variable Zoom screen enables display of the full variable name and path, and its value. To display the Variable Zoom screen for a row (variable), type option Z (Zoom) in the first column of the desired row in the Values of Database screen.

The Variable Zoom screen is displayed.

Figure 82 Variable Zoom Screen

### **Display Types of the Variable Zoom Screen**

The following predefined display types are available for the Variable Zoom screen.

**Table 102 Display Types of the Variable Zoom Screen** 

Туре	Description
D (Default display type)	Includes the first 64 characters of both the variable name and path, and the variable value for the selected database row.
	An additional line containing the remainder of the variable name and path (up to 76 characters), and an additional line containing the remainder of the variable value (up to 76 characters) can be displayed by option A (Additional Information), which is described in Table 103.
B (Blank Line display type)	This display type displays the second line for all variables and values, regardless of whether the line contains additional information.

#### **Changing Display Types**

While in the Variable Zoom screen, the display type can be changed using the DISPLAY command. Format of the command is:

DISPLAY X

where *x* is the identifying letter for the desired type.

DISPLAY can be abbreviated to DI.

#### **Example**

DI B

displays the Blank Line display type

### Uppercasing and lowercasing of variables' values

While in the Variable Zoom Screen, uppercase or lowercase mode can be set using the CAPS command. Format of the command is:

CAPS [{ON | OFF}]

In this command:

- CAPS ON Forces all user entries to be saved and displayed in uppercase characters, regardless of the case in which they were entered. Default.
- CAPS OFF Enables certain user entries to be saved and displayed in lowercase characters.
- CAPS Indicates whether CAPS ON or CAPS OFF mode is active.

#### NOTE



Name of variables do not support lowercase characters.

### **Options of the Variable Zoom Screen**

The following option is available in the Default display type of the Variable Zoom screen:

**Table 103 Options of the Variable Zoom Screen** 

Option	Description
Information)	Display a second line for the selected variable. This option can be entered for the VARNAME and/or the VALUE lines. If used, it displays the second line containing the remainder (up to 76 characters) of the value.

### **Exiting the Variable Zoom Screen**

- To exit the Variable Zoom screen and save the changes, press END (PF03/PF15).
- To exit the Variable Zoom screen without implementing the changes, press **RESET** (**PF04/PF16**), or type CANCEL in the COMMAND line and press **Enter**.

Changes made to the Variable database through the online Variable Database facility are not available to CONTROL-M or CONTROL-O until the modified database is reloaded into memory by the appropriate operator command (F CTMCMEM,LOADGLOBAL=IOAVAR), as described in the *INCONTROL* for z/OS Administrator Guide.

However, changes made to the Variable database through DO SET and SET VAR statements in CONTROL-M, and SET statements in the JCL, are kept in memory. The Variable database file is automatically updated during the next internal CONTROL-M interval cycle (or when the CONTROL-O or CMEM monitor is stopped.)

# **Condition and Resource Handling Facility**

Options 4 and 7 in the IOA Primary Option menu are directly related to the handling of IOA conditions and CONTROL-M resources. The screens displayed by these options are discussed on the following pages.

### **IOA Conditions/Resources Screen**

The IOA Conditions/Resources screen is accessed through Option 4 of the IOA Primary Option menu. It displays information from the IOA Conditions file, which contains the list of all existing prerequisite conditions, and the CONTROL-M Resources file, which contains the list of Quantitative resources and Control resources. The IOA Conditions/Resources screen enables you to

- view IOA prerequisite conditions
- view CONTROL-M Quantitative resources
- add or delete prerequisite conditions or resources or both
- change the available quantity of CONTROL-M Quantitative resources

For a description of prerequisite conditions, see "Prerequisite Conditions" on page 69

#### - NOTE



Prior to version 6.0.00 a single file, the IOA Conditions/Resources File, contained all IOA conditions and all Control and Quantitative resources. As of version 6.0.00, the IOA Conditions/Resources File has been replaced by two files:

- IOA Conditions file contains all IOA conditions
- CONTROL-M Resources file contains all Control and Quantitative resources

To enter the IOA Conditions/Resources screen, select Option 4 on the IOA Primary Option menu.

Figure 83 IOA Conditions/Resources Screen

(4)							
COMMAND ===> SCROII ===> CRSR							
PRFFIX ===>	COND Y	CONTROL	Y RF	S Y STAT	Υ	DATE 0606	
OPT TYPF	CONDITION/RESOURCE	IOAID		QUANTITY	MAX	*P RBA	DATE
CONTROL	CONTROLM	0.1	Ε	•		(00000)	
RESOURCE	TAPEP		В	0003	0003	( ,	
RESOURCE	CPU1		В	0098	0100		
RESOURCE	CPU2		В	0197	0200		
RESOURCE	TAPEP	01	U	0002		(00091)	
RESOURCE	CPU1	01	U	0002		(00091)	
RESOURCE	CPU2	01	U	0003		(00092)	
RESOURCE	TAPEP	01	R	0002		1 (00093)	
COND	BR-BRIVPCC-ENDED-OK						0909
COND	BR-BRCC0001-ENDED-OK						0909
COND	BR-BRCC0002-ENDED-OK						0909
COND	BR-BRCC0003-ENDED-OK						0909
COND	BR-BRCCIND-ENDED-OK						0909
COND	BR-BRUPDT02-ENDED-OK						0909
COND	BR-BRREPOO1-ENDED-OK						0909
COND	BR-BRREP002-ENDED-OK						0909
COND	GL-GLINPOO1-ENDED-OK						0909
COND	EBD-APPL-STARTED						0909
COND	CICS-PROD-IS-UP						STAT
OPTIONS: D DE	LETE C CHANGE ? WHY		COMMA	NDS: ADD		14.	07.08

To return to the IOA Primary Option menu, press the END key (PF03/PF15).

### Fields of the IOA Conditions/Resources Screen

The information displayed in each screen line is shown in Table 104:

Table 104 Fields of the IOA Conditions/Resources Screen (part 1 of 2)

Field	Description			
OPT	Option to be a	Option to be activated on the condition or resource.		
ТҮРЕ	Type of condition or resource:  COND – Prerequisite condition CONTROL – Control resource			
		E – Quantitative resource		
CONDITION/ RESOURCE	Name of the co	Name of the condition or resource.		
IOAID	Quantitative re	ID of the IOA installation that is using the particular Control or Quantitative resource. This value is significant when multiple IOA installations share the same resources.		
USE	Resource usage indicator for Control or Quantitative resources. Valid values depend on the type of resource.			
	For Control res	For Control resources, valid values are:		
		<ul> <li>■ E – Resource is being used in Exclusive mode</li> <li>■ S – Resource is being used in Shared mode</li> </ul>		
	For Quantitativ	For Quantitative resources, valid values are:		
	<ul> <li>B – Line indicates the initial definition for the resource</li> <li>U – Line indicates an instance of resource usage</li> <li>R – Line indicates an unfulfilled critical path request (that is, a request with an *-type priority) for the resource</li> </ul>			
QUANTITY		Quantitative resource. What the quantity represents e value in the USE field, as follows:		
	Use	Quantity		
	В	Quantity available. If the maximum quantity is more than 1 but only 1 is available, 0001 is displayed in pink for color terminals. If the maximum quantity is more than 1 but none is available, 0000 is displayed in red for color terminals.		
	U	Quantity in use by the particular process.		
	R	Quantity requested by the particular process, but unfulfilled.		
MAX	Maximum ava	Maximum available quantity of a Quantitative resource.		
*P	priority. For m	Priority of the job requesting a CONTROL-M resource using *-type priority. For more information, see "PRIORITY: Runtime Scheduling Parameter" on page 588.`		

Table 104 Fields of the IOA Conditions/Resources Screen (part 2 of 2)

Field	Description		
RBA	Internal CONTROL-M ID (relative byte address) of the job currently holding a CONTROL-M resource. An RBA value of 000000 indicates that the resource was added manually.		
	Notes:		
	■ Line indicates an unfulfilled critical path request (that is, a request with an *-type priority) for the resource.		
	■ It is permissible to add a duplicate CONTROL resource, either online in screen 4 or by using batch utility IOACND, to enable manual intervention in the job flow processing.		
DATE	Original date reference of a prerequisite condition (format mmdd or ddmm depending on the site standard, or the value STAT).		
	The conditions are displayed in day ( <i>dd</i> ) order (regardless of the site's date format), followed by STAT conditions.		

## **Specifying Retrieval Criteria**

You can control the type and amount of information displayed in the screen by specifying retrieval criteria.

Table 105 IOA Conditions/Resources Retrieval Criteria (part 1 of 2)

Field	Description		
PREFIX prefix	If specified, limits the display to conditions with the specified prefix.		
	To display only conditions containing a specific string, enter the string preceded by an *.		
	Example:		
	If *OK is entered, the following conditions are included in the display:		
	UPDATE - ENDED - OK OK - RUN OK		
COND	Determines whether prerequisite conditions are displayed. Valid values are:  ■ Y (Yes) – Display prerequisite conditions. Default. ■ N (No) – Do not display prerequisite conditions.		
CONTROL	Determines whether Control resources are displayed. Valid values are:  ■ Y (Yes) –Display Control resources. Default. ■ N (No) – Do not display Control resources.		

Table 105 IOA Conditions/Resources Retrieval Criteria (part 2 of 2)

Field	Description
RES	Determines whether Quantitative resources are displayed. Valid values are:  ■ Y (Yes) – Display Quantitative resources. Default.  ■ N (No) – Do not display Quantitative resources.
STAT	Determines whether prerequisite conditions with a date value of STAT are displayed. (Applies only if Y is specified for COND.) Valid values are:  ■ Y (Yes) – Include prerequisite conditions with a date value of STAT.  ■ N (No) – Do not include prerequisite conditions with a date value of STAT.
DATE from – to	Limits the display of prerequisite conditions to the selected date range. Valid values are:  ■ from − Earliest date in the date range, in mmdd or ddmm format (depending on the site standard). The default value is three days prior to the current date. This default can be modified in the Profile member by the INCONTROL administrator.  ■ to − Latest date in the date range, in mmdd or ddmm format (depending on the site standard). The default value is the current date.

### **Adding Conditions and Resources – The ADD Command**

From the Command field, use the ADD command to add conditions to the IOA Conditions file or resources to the CONTROL-M Resources file.

The format of the command is:

ADD type

where *type* is one of the following:

- COND Add a prerequisite condition. Special care must be taken when adding prerequisite conditions, because added conditions can trigger job submission.
- LCOND Add a long prerequisite condition (a condition name that exceeds 20 characters).
- RESOURCE/RES Add a Quantitative resource. Only authorized personnel should add Quantitative resources.
- CONTROL/CON Add a Control resource. A Control resource entry may be added manually even if a job is holding the resource. Only authorized personnel should add Control resources.

When the ADD command is entered, an appropriate window is opened. The window shown in Figure 84 opens when ADD COND is entered.

Figure 84 IOA Conditions/Resources COND Window

(4)					
COMMAN +					
PREFIX   PLEASE FILL IN COND NA	ME, DAT	E AND	PRESS ENTE	R	0606 - 0606
OPT TY					BA DATE
CO   NAME ===>		DD	MM ===>		00)
RE					
RE +				+	
RE					
RESOURCE TAPEP	01	U	0002	(	00091)
RESOURCE CPU1	01	U	0002	(	00091)
RESOURCE CPU2	01	U	0003	(	00092)
RESOURCE TAPEP	01	R	0002	1 (	00093)
COND BR-BRIVPCC-ENDED-OK					0606
COND BR-BRCC0001-ENDED-OK					0606
COND BR-BRCC0002-ENDED-OK					0606
COND BR-BRCC0003-ENDED-OK					0606
COND BR-BRCCIND-ENDED-OK					0606
COND BR-BRUPDT02-ENDED-OK					0606
COND BR-BRREPOO1-ENDED-OK					0606
COND BR-BRREP002-ENDED-OK					0606
COND GL-GLINPOO1-ENDED-OK					0606
COND EBD-APPL-STARTED					0606
COND CICS-PROD-IS-UP					STAT
OPTIONS: D DELETE C CHANGE ? WHY		COMMA	NDS: ADD		14.07.08

Fill in the window fields as described in Table 106 according to the specified ADD command:

**Table 106 IOA Conditions/Resources ADD Command Formats** 

Format	Description
ADD COND	Enter the name of the prerequisite condition. The current working date is displayed as the default date. This date can be modified.
ADD LCOND	Same as ADD COND, but for use only when the length of the condition name is from 21 through 39 characters.
ADD RESOURCE or ADD RES	Enter the name of the Quantitative resource and the quantity to be added.
ADD CONTROL or ADD CON	Enter the name of the Control resource and the control type (E – Exclusive; S – Shared).
	<b>Note</b> : If a Control resource is manually added with a type of E (Exclusive), no jobs in WAIT SCHEDULE status that require this resource are submitted.
	If a Control resource is manually added with a type of S (Shared), no jobs in WAIT SCHEDULE status that require exclusive access to this resource are submitted.

After filling in the window, press **Enter** to add the condition or resource.

To close the window without adding the condition or resource, press the **RESET** key (**PF04/PF16**).

### **Options of the IOA Conditions/Resources Screen**

The following options can be selected for conditions and resources by typing the option in the OPT field to the left of the resource or condition name and pressing **Enter**. Available options are shown in Table 107:

**Table 107 Options of the IOA Conditions/Resources Screen** 

Option	Description
D (DELETE)	Delete a condition or resource from the list. The event is recorded in the IOA Log file.
C (CHANGE)	Change the maximum available quantity of a Quantitative resource. The event is recorded in the IOA Log file.
? (WHY)	View a list of the jobs that are either waiting for this resource or holding it.

These options are discussed in detail on the following pages.

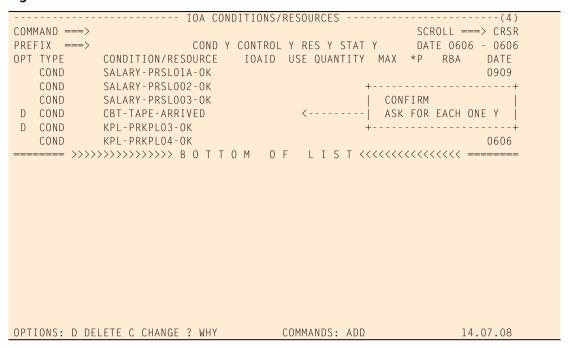
### **Deleting Conditions and Resources – The DELETE Option**

To delete conditions and/or resources, type **D** (Delete) in the **OPT** field to the left of the conditions and resources being deleted and press **Enter**.

A confirmation window may be displayed, depending on user profile customization:

- By default, conditions and resources are deleted without confirmation from the user.
- The User profile can be customized to display a confirmation window with an arrow pointing to a delete request (beginning with the first request).

Figure 85 IOA Conditions/Resources DELETE Confirmation Window



If a confirmation window is displayed, fill in the window as shown in Table 108 and press Enter:

**Table 108 IOA Conditions/Resources DELETE Confirmation Window Options** 

Option	Description
CONFIRM	Whether to process the delete request. Valid values are:  ■ Y (Yes) – Process the request. ■ N (No) – Cancel the request.
ASK FOR EACH ONE	This line is displayed only if more than one delete is requested. It determines whether individual confirmation is required for each delete request. Valid values are:  Y (Yes) – Individual confirmation is required for each delete request. The specified CONFIRM value applies only to the current order or request.
	N (No) – Individual confirmation is not required for each delete request. The specified CONFIRM value is applied to all delete requests. (If CONFIRM is Y, all delete requests are processed; if CONFIRM is N, no delete request is processed.)

#### Changing the Quantity of a Resource – The CHANGE Option

To request a change to the maximum available quantity of a resource, type C (Change) in the OPT field to the left of the resource and press **Enter**. The window shown in Figure 86 is opened.

Figure 86 IOA Conditions/Resources CHANGE Option Window

IOA CONI	DITIONS/	RESOU	RCFS		(4)	
COMMAN +					+ L ===> CRSR	
PREFIX   PLEASE FILL IN QUANT RES	NAME, C	COUNT	AND PRESS E	ENTER	0606 - 0606	
OPT TY					BA DATE	
CO NAME ===> TAPEP		CO	UNT ===>		000)	
C RE						
RE +					+	
RE						
RESOURCE TAPEP	01	U	0002		(00091)	
RESOURCE CPU1	01	U	0002		(00091)	
RESOURCE CPU2	01	U	0003		(00092)	
RESOURCE TAPEP	01	R	0002	1	(00093)	
COND BR-BRIVPCC-ENDED-OK					0606	
COND BR-BRCC0001-ENDED-OK					0606	
COND BR-BRCC0002-ENDED-OK					0606	
COND BR-BRCC0003-ENDED-OK					0606	
COND BR-BRCCIND-ENDED-OK					0606	
COND BR-BRUPDT02-ENDED-OK					0606	
COND BR-BRREPOO1-ENDED-OK					0606	
COND BR-BRREP002-ENDED-OK					0606	
COND GL-GLINPOO1-ENDED-OK					0606	
COND EBD-APPL-STARTED					0606	
COND CICS-PROD-IS-UP					STAT	
OPTIONS: D DELETE C CHANGE ? WHY		COMMA	NDS: ADD		14.07.08	

The NAME value in the window is protected and cannot be changed.

The COUNT parameter consists of two values: sign and quantity. Fill in the COUNT parameter as shown in Table 109 and press **Enter**:

**Table 109 COUNT Parameter Values** 

Value	Description
sign	Valid values (one character):  ■ + (Plus) – Add the selected quantity to the current maximum available quantity to give a new maximum available quantity.
	<ul> <li>- (Minus) – Subtract the selected quantity from the current maximum available quantity to give a new maximum available quantity.</li> </ul>
	<ul><li>' ' (Blank) – Set the maximum available quantity to the selected quantity.</li></ul>
quantity	Quantity to be used to adjust the maximum quantity of the resource (four digits) according to the specified sign. Leading zeros are required.

#### Viewing a List of the Jobs Waiting for a Resource – The WHY Option

To view a list of the jobs that are either waiting for this (quantitative or control) resource or holding it, type? in the OPT field to the left of the resource and press **Enter**. The window shown in Figure 87 is displayed:

Figure 87 Resource Analysis WHY Option Window

The information displayed for each job in the list is shown in Table 110:

Table 110 Fields of the WHY Option Window

Field	Description
MEMBER	The job's member name
RBA	The job's relative byte address
ORDERID	The job's Order ID
PRIORITY	The job's priority field
QUANTITY	The number of Quantitative resources held by this job. If the resource being viewed is not quantitative, then this field will contain blanks
USE	The mode in which the control resource is used by this job. Valid values are:  E – the control resource is used in Exclusive mode  S – the control resource is used in Shared mode  ' ' (Blank) – the resource is not a control resource
DISP	Whether the resource is to be kept, released, or discarded when the job finishes running. The field contains two 1-character values, separated by a comma. The first value represents the behavior if the job ends normally, and the second value represents the behavior if the job fails. Valid values are:  R - the resource will be released K - the resource will be kept D - the resource will be discarded`
STATUS	Whether this job is currently waiting for the resource, or holding it. Possible values are HOLDING and WAITING.

### **IOA Manual Conditions Screen**

The IOA Manual Conditions screen displays a list of prerequisite conditions that must be confirmed manually by operations personnel.

The list of manual conditions is created by utility IOALDNRS. The utility is described in the *INCONTROL for z/OS Utilities Guide*. This utility is intended for use at sites where CONTROL-M and/or CONTROL-D are installed.

The utility scans the jobs in the CONTROL-M Active Jobs file, and/or missions in the CONTROL-D Active Missions file, for all conditions requested in IN statements that

- are not resolved by an OUT statement
- are not resolved by ON PGMST or DO COND statements
- do not exist in the IOA Conditions file

The utility automatically places the conditions conforming to the above criteria into the IOA Manual Conditions file. This file is used as a checklist for manual operations that operations personnel are expected to perform.

To enter the IOA Manual Conditions screen, select Option 7 on the IOA Primary Option menu.

Figure 88 IOA Manual Conditions Screen

```
-----(7)
COMMAND ===>
                                                    SCROLL ===> CRSR
PREFIX ===>
                        PENDING Y ADDED Y STAT Y
                                                    DATE 0606 - 0606
OPT TYPE
COND
           CONDITION
                                               DATE ADDED
           USR-GOT-TAX-TAPE
                                               0606
   COND
           DBA-RUN-UPDATE
                                               0606
                                                      Υ
   COND
           OP-EXTERNAL-TAPE-OK
                                               0606
                                                      Υ
   COND
           USR-GOT-BANK-TAPE
                                               0606
   COND
           OP-SHUT-THE-SYSTEM
                                               0606
   COND
           DBA-START-MPMXXX
                                               0606
   COND
           USR-GOT-SALARY-TAPE
                                               0606
   COND
           OP-COMMUNICATION-DOWN
                                               0606
  ==== >>>>>>> B O T T O M O F L I S T <<<<<<<<< <-----
OPTIONS: A ADD TO COND/RES LIST (SCREEN 4) E ERASE
                                                COMMANDS: NEW 18.33.47
```

To exit the IOA Manual Conditions screen, press END (PF03/PF15).

### Fields of the IOA Manual Conditions Screen

The information displayed on each screen line is shown in Table 111:

**Table 111** Fields of the IOA Manual Conditions Screen

Field	Description
OPT	Option to be activated on the condition.
TYPE	Type of condition, meaning, COND for prerequisite condition.
CONDITION	Condition name.
DATE	Date reference of prerequisite condition. Format is either mmdd or ddmm depending on the site standard, or the date value STAT.
ADDED	Indicates whether the condition has been manually added to the IOA Conditions file.  ■ N (No) – Condition has not been added. ■ Y (Yes) – Condition has been added.

# **Specifying Retrieval Criteria**

You can control the type and amount of information displayed in the screen by specifying retrieval criteria.

Table 112 Retrieval Criterion for IOA Manual Conditions Screen (part 1 of 2)

Criterion	Description
PREFIX prefix	Limits the display to conditions with the selected prefix. Default: Blank (no limit).
	To display only those conditions containing a specific string, enter the string preceded by an *.
	Example:
	If *OK is entered, the following conditions are included in the display:
	UPDATE-ENDED-OK OK-RUN OK
PENDING	Determines whether conditions not yet added are displayed. Valid values are:  ■ Y (Yes) – Display pending conditions. ■ N (No) – Do not display pending conditions.
ADDED	Determines whether added conditions are displayed. Valid values are:  ■ Y (Yes) – Display added conditions.  ■ N (No) – Do not display added conditions.

Table 112 Retrieval Criterion for IOA Manual Conditions Screen (part 2 of 2)

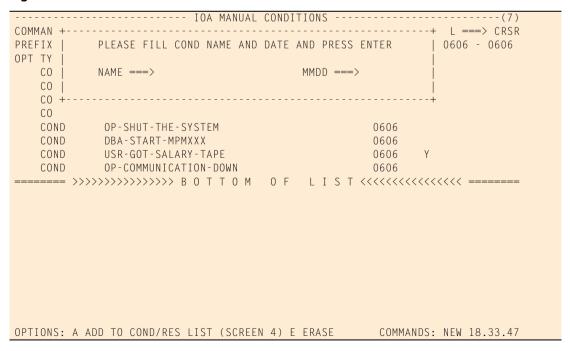
Criterion	Description
STAT	<ul> <li>Determines whether prerequisite conditions with a date value of STAT are displayed. Valid values are:</li> <li>■ Y (Yes) – Display prerequisite conditions with a date value of STAT.</li> <li>■ N (No) – Do not display prerequisite conditions with a date value of STAT.</li> </ul>
DATE from – to	Limits the display of prerequisite conditions to the selected date range. Valid values are:  ■ from − Earliest date in the date range, in mmdd or ddmm format (depending on the site standard). The default value is three days before the current date.  ■ to − Latest date in the date range, in mmdd or ddmm format (depending on the site standard). The default value is the current date.

### **Adding a New Prerequisite Condition – NEW Command**

To add a prerequisite condition to the IOA Manual Conditions file, type **NEW COND** in the COMMAND field and press **Enter**. The window shown in Figure 89 is opened.

To add a condition with a name from 20 through 39 characters in length, type NEW LCOND in the COMMAND field and press **Enter**. The window shown in Figure 89 is opened.

Figure 89 IOA Manual Conditions Screen NEW Window



In the NAME field of the window, type the name of the condition to be added. If the condition has a date other than the current working date, enter the date in the MMDD field of the window, in the format DDMM or MMDD, depending on the site standard.

- To add the condition, press **Enter**.
- To close the window without adding the condition, press **RESET** (**PF04/PF16**).

#### NOTE



Adding a new condition to the IOA Manual Conditions file does not affect the IOA Conditions file.

### **Options of the IOA Manual Conditions Screen**

To add a condition to the IOA Conditions file, or to erase a condition from the IOA Manual Conditions file, type the appropriate option in the **OPT** field to the left of the condition name and press **Enter**. Valid options are shown in Table 113:

**Table 113 Options of the IOA Manual Conditions Screen** 

Option	Description
A (ADD)	Add the condition to the IOA Conditions file (screen 4), and mark it "Added" (Y) in the IOA Manual Conditions file. The event is recorded in the IOA Log file.
E (ERASE)	Erase (Delete) a condition from the IOA Manual Conditions file. This does not affect the IOA Conditions file. This option is discussed in more detail below.

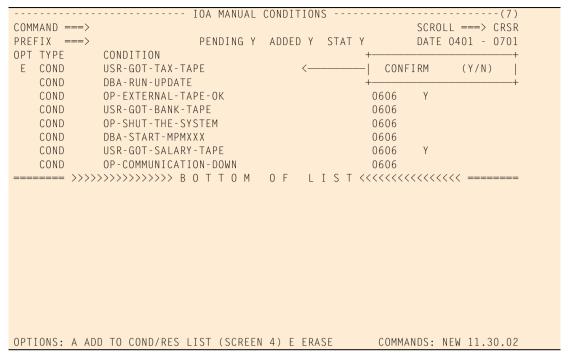
### **Erasing (Deleting) Conditions**

To erase prerequisite conditions, type **E** in the **OPT** field to the left of the condition names being erased and press **Enter**.

A confirmation window may be displayed, depending on user profile customization:

- By default, conditions are deleted without confirmation from the user.
- If, however, the user profile member has been customized accordingly, a confirmation window is displayed with an arrow pointing to an erase request (beginning with the first request).

Figure 90 IOA Manual Conditions Screen ERASE Confirmation Window



If a confirmation window is displayed, fill in the window as follows and press **Enter**:

Table 114 Fields of the IOA Manual Conditions Screen ERASE Confirmation Window

Field	Description
CONFIRM	Indicates whether to process the erase (delete) request. Valid values are:  ■ Y (Yes) – Process the request.  ■ N (No) – Cancel the request.
ASK FOR EACH ONE	This line is displayed only if more than one erase is requested. It determines whether individual confirmation is required for each erase request. Valid values are:  ■ Y (Yes) – Individual confirmation is required for each erase request. The selected CONFIRM value applies only to the current order or request.  ■ N (No) – Individual confirmation is not required for each erase request. The selected CONFIRM value is applied to all erase requests. If CONFIRM is Y, all erase requests are processed; if CONFIRM is N, no erase request are processed.

# **IOA Log Facility**

The IOA Log facility places automatically generated messages, which record every significant event in the life of a job, rule or mission, in the IOA Log file. Significant events recorded in the IOA Log file include normal processing occurrences, such as job submitted, as well as error conditions encountered during processing, such as job abends. Shout facility notifications and user remarks may also be recorded in the IOA Log file.

# **IOA Log Screen**

The IOA Log screen enables you to view the information contained in the Log file. To enter the IOA Log screen, select option 5 on the IOA Primary Option menu. Upon entry, the screen displays the most recent messages currently in the IOA Log file.

Figure 91 IOA Log Screen

FILTER	IOA LOG(5)
COMMAND ===>	SCROLL===> CRSR
SHOW LIMIT ON ==>	DATE 060601 - 060601
DATE TIME ODATE USERID	CODE M E S S A G E
060601 092144 060601 M22	SPY254I JOB CT085955 CT085955/01835 SCANNED
060601 092144 060601 M22	SEL208I JOB CT085955 CT085955/01835 ENDED "OK"
060601 092150 060601 M22	SPY254I JOB CT085956 CT085956/01836 SCANNED
	SEL208I JOB CT085956 CT085956/01836 ENDED "OK"
	SPY254I JOB BRIVPCC BRIVPCC /01843 SCANNED
	SEL208I JOB BRIVPCC BRIVPCC /01843 ENDED "OK"
060601 092157 060601 DBA	CTM659I FREE OF TASK BRCC0001 ODATE 060601
	PERFORMED
060601 092201 060601 M22	SPY281I JOB INTROOO4 INTROOO4/04371 START
	98253.1316 STOP 98253.1316 CPU OMIN
0.0001 000001 000001 M00	00.04SEC SRB OMIN 00.00SEC 0.19
	SPY254I JOB INTROOO4 INTROOO4/04371 SCANNED
060601 092201 060601 M22	SEL206W JOB INTROOO4 INTROOO4/04371 ABENDED CC SB37 STEP STEP01
060601 002201 060601 M22	SEL219I JOB INTROOO4 INTROOO4/04371 ENDED "NOT
000001 092201 000001 M22	OK"
060601 092208 060601 TVP	SEL203I JOB BRCC0001 ELIGIBLE FOR RUN
	SUB133I JOB BRCC0001 BRCC0002/01958 SUBMITTED
	SEL203I JOB BRCC0002 ELIGIBLE FOR RUN
CMDS: SHOW, GROUP, CATEGORY,	

To return to the IOA Primary Option menu, press END (PF03/PF15).

# Fields of the IOA Log Screen

Table 115 Fields of the IOA Log Screen

Field	Description
SHOW LIMIT ON	Identifies which selection criteria other than yes or no were entered in the IOA Log Show Screen window (USERID, MEM/MIS, JOBNAME, CATEGORY, GROUP). For more information, see "Filtering the IOA Log Screen Display" on page 299.
DATE	Date on which the message was issued.
TIME	Time at which the message was issued.
ODATE	Original scheduling date of the job associated with the message. Format is mmddyy, ddmmyy or yymmdd, depending on the site standard.
	Note: When the display type is set to RBA display using the DISPLAY command, the Relative Byte Address (RBA) of the message within the IOA Log file is displayed instead of the ODATE. For more information on changing the screen display, see "Changing IOA Log Screen Display Types" on page 298.
USERID	User ID of the job issuing the message, or user ID of the user writing to the log.
CODE	IOA message code.
MESSAGE	Text of the message. If the message is longer than the space available on the screen, the message is split and continues on the following line. Messages relating to a job have the following format:  tasktype memname jobname/jobid message
fromdate – todate	Log information displayed in the screen can be limited to the specified date range in mmddyy, ddmmyy or yymmdd format, depending on the site standard. If the DATE or the ODATE value for a message is in the range selected, the message is included in the IOA Log display. Valid values are: <ul> <li>fromdate – Earliest date in the date range.</li> <li>todate – Latest date in the date range.</li> </ul>

# **Commands of the IOA Log Screen**

The following commands can be entered in the COMMAND field.

Table 116 Commands of the IOA Log Screen

Command	Description
SHOW	The SHOW command activates the specified screen filter, opens the IOA Log Show Screen window, or opens the Display Filters window, depending on the format of the command. For more information on filtering the IOA Log Screen, see "Filtering the IOA Log Screen Display" on page 299. Valid formats are:
	■ SHOW <i>name</i> – Activates the specified filter.
	<ul> <li>SHOW? – Opens the Display Filters window, which lists all available filters.</li> </ul>
	■ SHOW ( <b>PF02/PF14</b> ) – Opens the IOA Log Show Screen window for the currently active filter, and allows editing of that filter.
	■ SHOW name EDIT – Opens the IOA Log Show Screen window for the specified filter, and allows editing of that filter.
	<b>Note:</b> Reserved filter name DEFAULT can be used to activate or edit the default filter for the status screen. For example, SHOW DEFAULT EDIT opens the IOA Log Show Screen window for the default filter.
	Only jobs conforming to selection criteria specified in the filter are displayed in the IOA Log screen. For more information, see "Filtering the IOA Log Screen Display" on page 299.
GROUP	The GROUP Command alternately displays or hides the GROUP name (if any) that is associated with the relevant job, mission or rule definition. When displayed, the name of the group appears after the job, mission or rule status.
CATEGORY	The CATEGORY command alternately displays and hides the CATEGORY of the relevant CONTROL-D mission. This command applies to CONTROL-D generated messages only. When displayed, the name of the category appears after the mission status.
	Note: At sites where CONTROL-D is active.

# **Changing IOA Log Screen Display Types**

While in the IOA Log screen, the display type can be changed through the DISPLAY command. The format of the command is:

#### DISPLAY x

where *x* is a 1-character code that identifies the desired display type. DISPLAY can be abbreviated DI.

#### **NOTE**



For a list of display types, enter **DISPLAY**? to show the Display Options window. To select a display type in the window, type **S** in the Option field next to the ID. To exit the window without selecting a display type, press the **END** key (**PF03/PF15**).

#### **Example**

DI B

displays the No Reverse display.

Valid predefined displays are shown in Table 117.

**Table 117 IOA Log Screen Predefined Display Types** 

Display	Description
A	Show RBA (Relative Byte Address) display (displays the RBA of the message within the IOA Log file in place of the ODATE)
D	Default display
В	No Reverse display (display is in No Reverse mode)

### Filtering the IOA Log Screen Display

Screen filters can be used to filter the IOA Log screen display.

A filter consists of a set of record selection criteria (selection fields and their values). Only records that conform to the selection criteria specified in the filter are displayed on the screen.

The INCONTROL administrator can predefine filters and place them in the General profile.

Each user can activate an existing filter in the IOA Log screen by entering the SHOW command in the COMMAND line of the IOA Log screen.

Each user can define multiple filters for the screen, through the IOA Log Show Screen window, which is described in "Fields of the IOA Log Show Screen Window" on page 302. User-defined filters belong to, are assigned names by, and can only be activated by, the user who defined them. They are stored in the user profile.

You can see the list of all available filters by opening the Display Filters window.

A predefined default filter (DEFAULT) is defined for the IOA Log screen. Site-defined defaults determine whether the last filter used or the DEFAULT filter is activated upon reentry to the IOA Log screen.

#### Activating an existing filter in the IOA Log screen

The SHOW command can be used to activate an existing filter when you know the name of the filter.

■ To activate an existing filter in the IOA Log screen, enter the SHOW command in the COMMAND field, as follows:

SHOW name

where *name* is the name of the filter to be activated.

■ To activate the DEFAULT filter, use DEFAULT as the name of the filter.

### **Display Filters Window**

When you do not know the name of a filter, you can still activate a filter from the list of available filters by opening the Display Filters window. This window displays the list of all available filters. These include Global filters that are available to all users, as well as user-defined filters that are only available to the individual user. You can activate a filter from the Display Filters window, or switch to the IOA Log Show Screen window, where you can edit or define a filter.

To enter the Display Filters window, type SHOW ? in the **COMMAND** field of the IOA Log screen and press **Enter**.

Figure 92 IOA Log Screen Display Filters Window

```
FILTER: DEFAULT
                 -----(5)
                                                             SCROLL==> CRSR
COMMAND ===>
SHOW LIMIT ON ==>
                                                        DATE 060601 - 060601
DATE TIME ODATE USERID CODE ----- M E S S A G E -----
060601 092144 060601 M22 SPY254I JOB CT085955 CT085955/01835 SCANNED SEL208I JOB CT085955 CT085955/01835 ENDED "OK"
       -----+ B CT085956 CT085956/01836 SCANNED
      DISPLAY FILTERS | B CT085956 CT085956/01836 ENDED "OK"
0
  CMD ==> SCROLL==> CRSR | B BRIVPCC BRIVPCC /01843 SCANNED | O NAME DESCRIPTION | B BRIVPCC BRIVPCC /01843 ENDED "
0
                                     B BRIVPCC BRIVPCC /01843 ENDED "OK"
0
0
      CONFIRM
                                      EE OF TASK BRCCOOO1 ODATE 080800
      DEL
                                      RFORMED
0
                                      B INTRO004 INTRO004/04371 START
      END
       ENDNOTOK
                                      253.1316 STOP 98253.1316 CPU OMIN
       ENDOK
                                      .04SEC SRB OMIN 00.00SEC 0.19
       EXEC
                                      B INTROOO4 INTROOO4/04371 SCANNED
0
       LATE
                                      B INTROOO4 INTROOO4/04371 ABENDED CC
       WAIT
                                      37 STEP STEP01
0
                                      B INTROOO4 INTROOO4/04371 ENDED "NOT
    =====>>> BOTTOM <<<======
0
                                      B BRCC0001 ELIGIBLE FOR RUN
                                    B BRCC0001 BRCC0002/01958 SUBMITTED
0
    OPTIONS S SELECT E EDIT
                               ----+ B BRCC0002 ELIGIBLE FOR RUN
CMDS: SHOW, GROUP, CATEGORY, SHPF
                                                                    09.43.00
```

#### **Fields of Display Filters Window**

The Display Filters window contains the fields shown in Table 118:

**Table 118 Fields of the Display Filters Window** 

Field	Description
NAME	Name of the filter as it appears in the General or user profile.
DESCRIPTION	Description of the filter.

#### **Options of the Display Filters Window**

To request one of the following options, type the option in the **OPT** field to the left of the filter name and press **Enter**.

Table 119 Options of the Display Filters Window

Option	Description
S (SELECT)	Filters the entries that are displayed in the Automation Log Screen according to the criteria specified in the selected filter.
E (EDIT)	Opens the IOA Log Show Screen window, where the filter criteria are displayed. You can modify the filter criteria.

### **IOA Log Show Screen Window**

The IOA Log Show Screen window in the IOA Log screen enables you to create or modify a filter.

■ To open an existing filter for editing, enter the following command:

```
SHOW filtername EDIT
```

where filtername is the name of the filter to be displayed in the IOA Log Show Screen window.

■ To edit the currently active filter, it is not necessary to enter the name of the filter or the EDIT keyword. Enter the SHOW command in the COMMAND field, or press **PF02/PF14**. The following IOA Log Show Screen window is displayed:

Figure 93 IOA Log Show Screen Window

FILTER: DEFA	+(5)
COMMAND ===>	FILTER DEFAULT SAVE (Y/N) DESC:
SHOW LIMIT 0	CM : D JOB M JOB SHOUT USER GENERAL D INT M INT STAT
DATE TIME	j yy y n n n y j
060601 14131	CMEM : GENERAL
	i i
	i
060601 14131	

060601 14131	
060601 14153	
060601 14155	
	CODE
	URGENCY: REGULAR Y URGENT Y VERY-URGENT Y
060601 14155	TASK TYPE CM: JOB CYC EMR STC CST EST ECJ ECS WRN GRP
060601 14155	Y Y Y Y Y Y Y Y Y
060601 14173	
060601 14174	USERID N54A
	MEM/MIS MIGDASD
	JOBNAME
060601 14174	CATEGORY
060601 14174	GROUP
060601 14174	ORDERID
CMDS: SHOW,	·

■ To create a new filter, open any existing filter and enter a new name and description in the FILTER and DESC fields (described in "Fields of the IOA Log Show Screen Window," below).

#### Fields of the IOA Log Show Screen Window

The IOA Log Show Screen window contains the following fields:

Table 120 Fields of the IOA Log Show Screen Window

Field	Description
FILTER	User-assigned name of the filter. The name entered in the FILTER field can be modified.
	If changes to a filter have not been saved, an asterisk is displayed to the right of the filter name. For more information, see "Closing the IOA Log Show Screen Window" on page 305.
SAVE (Y/N)	Specifies whether to save modifications to the filter upon closing the window.
Desc:	User-defined description of the filter. The description entered here appears next to the name in the Displaying Filters window.

#### - NOTE -



The INCONTROL administrator can limit which installed INCONTROL products and options each user may access. However, because all INCONTROL products and the messages they issue are integrated, it may be important for users to see the messages of products and options to which they have no access. Therefore, the types of messages for all INCONTROL products are listed in the IOA Log Show Screen window, and by default, the messages of all installed products are listed in the IOA Log screen.

Fields that define the selection criteria to be applied to the screen are described below. Fill in the selection criteria as necessary.

#### – NOTE



The selection criteria marked with the  P  symbol act on a prefix basis. For example, typing D4 in the JOBNAME field causes the retrieval of all jobs whose names start with D4.

Table 121 IOA Log Show Screen Window Selection Criteria (part 1 of 2)

Criteria	Description
CM message type	To limit the type of log messages displayed, enter Y (Yes) or N (No) under the desired message type.  Valid CONTROL-M message type codes are:  D JOB – Messages related to jobs produced during New Day processing.  M JOB – Job-related messages produced by the CONTROL-M monitor. The majority of job messages are of this type.  SHOUT – Messages written to the Log file by the SHOUT parameter. For more information, see "SHOUT: Post–Processing Parameter" on page 627.  USER – Messages resulting from manual intervention of authorized users in the operation of CONTROL-M; for example, the addition of a prerequisite condition, HOLD or RERUN of the job, and so on.  GENERAL – General messages on CONTROL-M operation.  D INT – Internal messages generated during New Day processing. For use mainly by maintenance personnel.  M INT – Certain CMEM messages, and internal messages of the CONTROL-M monitor.
CMEM message type	To limit the type of log messages displayed, enter Y (Yes) or N (No) under the desired message type. Valid CMEM message type code:  GENERAL – General messages on CMEM operation.
specific to CONTROL	a identified by "CM message type" and "CMEM message type" are -M and CMEM, respectively. Other selection criteria, such as those orimarily applicable to other INCONTROL products, but may also be L-M and CMEM.
CODEP	Show only IOA Log file messages with the specified message IDs or prefix(es). A maximum of 6 message IDs (or prefixes) can be specified.
URGENCY	Mark Y (Yes) or N (No) to specify the desired urgency of messages. Urgent and very urgent messages are highlighted.

**Table 121 IOA Log Show Screen Window Selection Criteria (part 2 of 2)** 

Criteria	Description
TASK TYPE	When job messages are selected, limit the task types to be displayed. Type Y to include or N to exclude the following task types:  JOB – Regular job. CYC – Cyclic job. EMR – Emergency job. STC – Started task. CST – Cyclic started task. EST – Emergency started task. ECJ – Emergency cyclic job. ECS – Emergency cyclic started task. WRN – Warnings. Supported for historical reasons only. GRP – Group Entity.
USERID <i>P</i>	Show only messages of the selected user IDs. A maximum of five user IDs can be specified.
the display of message these selection criteria	MEM/MIS, JOBNAME, and GROUP, described below, only affect is related to a job. Messages not related to a job are not affected by and are displayed unless suppressed by other selection criteria.
MEM/MISP	Limit displayed job messages to the selected member names. A maximum of five member names can be specified. Messages not related to a job are not affected by this show limit.
JOBNAME <i>P</i>	Limit displayed job messages to the selected job names. A maximum of five job names can be specified. Messages not related to a job are not affected by this show limit.
CATEGORY	CONTROL-D category. This selection field is not relevant to CONTROL-M and does not filter CONTROL-M jobs.
GROUPP	Limit displayed job messages to the selected groups. A maximum of four groups can be specified. Messages not related to a job are not affected by this show limit.
ORDERID	Limit displayed job messages to the specified order ID

# IOA Log Show Screen window (at Sites Where Multiple IOA Products Are Active)

The IOA Log Show Screen window displays different selection criteria depending on which INCONTROL products are operational at your site.

The IOA Log Show Screen window at sites where all INCONTROL products are active is illustrated in Figure 94.

Figure 94 IOA Log Show Screen Window at Sites where Multiple INCONTROL Products are Active

FILTER: DEFA	+(5)
COMMAND ===>	FILTER SAVE (Y/N)
SHOW LIMIT O	CM : D JOB M JOB SHOUT USER GENERAL D INT M INT STAT
DATE TIME	Y Y Y Y N N N
060800 21354	CO+CMEM: GENERAL SHOUT JOBS GENERAL W PIPE W JOB W   Y Y Y
060601 22040	CD+CV : SBSYS REP MIS SHOUT USER GENERAL DAILY MONIT STAT
060601 22040	Y Y Y Y Y N N N
060601 22040	CB : RUNTIME SHOUT DAILY GENERAL STATISTICS
	Y Y Y Y
060601 22040	CT : GENERAL SHOUT REAL-TIME UTILITIES
060601 22040	Y Y Y
060601 22040	CODE
	URGENCY: REGULAR Y URGENT Y VERY-URGENT Y
060601 23034	TASK TYPE CM: JOB CYC EMR STC CST EST ECJ ECS WRN GRP
060601 23040	
060601 23040	CD: REP PRT BKP/MIG RST EMR NOEMR CYC NOCYC
060601 23040	Y Y Y Y Y Y
	USERID N54A
060601 23040	MEM/MIS MIGDASD
060601 23040	JOBNAME
060601 23040	CATEGORY
	GROUP
	ORDERID
CMDS: SHOW,	++

The CONTROL-M selection criteria are described in Table 121 on page 303. For descriptions of the selection options for other INCONTROL products, see the user guides of the respective products.

#### **NOTE**



The INCONTROL administrator can limit which installed INCONTROL products and options each user can access. However, because all INCONTROL products and the messages they issue are integrated, it may be important for users to see the messages of products and options to which they have no access. Therefore, the types of messages for all INCONTROL products are listed in the IOA Log Show Screen window, and by default, the messages of all installed products are listed in the IOA Log screen.

### **Closing the IOA Log Show Screen Window**

You can activate an edited filter with or without saving changes, depending on the value you type in the **SAVE** field, as follows:

- To activate and save the filter, type Y (Yes) in the SAVE field. Changes to the filter are permanently saved.
- To activate the filter without saving it, type N (No) in the SAVE field. Changes are kept in memory only, but are not saved.

After entering a value in the **SAVE** field, press one of the following keys:

Table 122 IOA Log Show Screen window - Closing Values

Key	Description
Enter	Filtering begins with the first message currently displayed in the screen and continues downward.
<b>PF07</b> (UP)	Filtering begins with the first message in the IOA Log file and continues downward.
PF08 (DOWN)	Filtering begins with the last message in the IOA Log file and continues upward.

The window is closed and the filter is activated as defined or modified.

To cancel changes made in the IOA Log Show Screen window, press **RESET** (**PF10/PF22**). The changes are canceled regardless of the value entered in the **SAVE** field, the window is closed, and the filter that was previously in effect is restored.

By default, pressing **END** (**PF03/PF15**) in the window works like pressing **Enter**. However, the default can be modified so that pressing **END** works like pressing **RESET**.

# **IOA Calendar Facility**

The IOA Calendar facility enables you to create, view, or modify calendar definitions.

Calendars simplify the scheduling of INCONTROL product jobs, missions, rules, and so on. When a particular schedule is used in many job scheduling, mission, and/or rule definitions, a calendar can be defined for that schedule, and the name of that calendar can be specified in all the job, mission, or rule definitions that use that particular schedule.

For example, calendars may be defined to handle the normal scheduling needs for workdays, holidays, weekends, beginning of month, end of month, and so on. Exception calendars may also be created.

A calendar definition consists of parameters that specify when scheduling occurs.

Calendar definitions are stored in members. A member usually contains multiple calendar definitions, as follows:

A member contains the calendars required for a specific type of scheduling need. For example, the calendar member WORKDAYS may contain the calendar definitions for normal workday scheduling. ■ Each calendar definition in that member defines the schedule for a given year. For example, the calendar member WORKDAYS may contain calendar definitions 2001, 2002, and 2003. Each of those definitions contains the normal workday schedule for the corresponding year.

The IOA Calendar facility also enables the definition of varied work periods throughout the year, in special calendars called periodic calendars.

A calendar definition needs to be created only once. Once defined, the definition is saved and used as necessary for scheduling. Calendar definitions can be modified or deleted as required.

Any number of calendar members can be defined. Calendar members are stored in calendar libraries (partitioned data sets). Generally one calendar library is defined at time of installation, and referenced by the DACAL DD statement.

#### – NOTE



The IOA Calendar facility does not support members that have been compressed using the ISPF PACK option.

Ensure that each IOA calendar that you define contains entries for the years preceding and following the period you want to define. The entries for the preceding and following years can be dummy entries, provided that these years contain at least one day set to Y. If an IOA calendar does not contain entries for the preceding and following years, problems may arise when CONTROL-M resolves scheduling criteria, in which event the CTM707E message is displayed.

# Accessing the IOA Calendar Facility

The IOA Calendar facility contains the screens shown in Table 123:

**Table 123 IOA Calendar Facility Screens** 

Screen	Description
IOA Calendar Facility entry panel	Enables specification of parameters that determine which records are displayed in subsequent screens.
Calendar List screen	Displays the list of calendar members in the selected calendar library.
Year List screen	Displays the list of years for which there is a calendar definition in the selected calendar member.
Calendar Definition screen	Displays the parameters of the selected calendar for the selected year. This is the main screen of the facility.

To enter the Calendar facility, select Option 8 in the IOA Primary Option menu. The Calendar facility entry panel is displayed.

Depending on the values entered in the entry panel, you can bypass the Calendar List screen and/or the Year List screen.

# **Entry Panel**

The entry panel is displayed upon entering the IOA Calendar facility (Option 8 in the IOA Primary Option menu).

Figure 95 IOA Calendar Facility – Entry Panel

```
COMMAND ===>

SPECIFY LIBRARY, CALENDAR, YEAR

LIBRARY ===> IOA.PROD.CAL
CALENDAR ===> (Blank for calendar selection list)
YEAR ===> (Blank for year selection list)

USE THE COMMAND "SHPF" TO SEE PFK ASSIGNMENT 10.58.42
```

### **Fields of the Entry Panel**

Fill in the fields shown in Table 124 and press Enter.

Table 124 Fields of the IOA Calendar Facility Entry Panel

Field	Description
LIBRARY	Name of the desired calendar library. Mandatory.
	If you make an entry in this field without filling in the <b>CALENDAR</b> field, the list of calendars in the selected library is displayed in the Calendar List screen.
	If you make an entry in this field, you can restrict the list of calendars that are displayed by entering in the <b>CALENDAR</b> field part of a Calendar name together with a mask character or characters (? and *).
CALENDAR	Name of the desired calendar member. Optional.
	If you make an entry in this field without filling in the <b>YEAR</b> field, the list of years in the selected calendar member is displayed in the Year List screen.
YEAR	Year of the desired calendar definition. Optional.
	This field can be used only if a CALENDAR value is also entered. If specified, the calendar definition is displayed in the Calendar Definition screen.

#### NOTE -



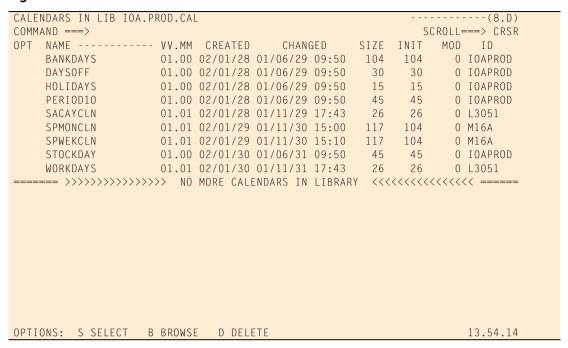
If you use the selection list fields, their values are not erased until you exit the entry panel by pressing  $END\ (PF03/PF15)$ .

### **Calendar List Screen**

The Calendar List screen displays a list of calendars (members) in the selected library. This screen can be entered directly from the entry panel or upon exiting the Year List screen.

By default, only calendar names are listed in the screen. However, if the default has been modified at time of installation, statistical information is displayed for each calendar name, as shown in Figure 96.

Figure 96 Calendar List Screen



To return to the entry panel, press END (PF03/PF15).

### **Options of the Calendar List Screen**

To request one of the following options, type the option in the **OPT** field to the left of the calendar names, and press **Enter**.

**Table 125 Options of the Calendar List Screen** 

Option	Description
S (SELECT)	Display the list of years for the calendar for any purpose, including editing or modification. Only one calendar can be selected at a time.
B (BROWSE)	Display the list of years for the calendar for browsing. Only one calendar can be selected at a time.
D (DELETE)	Delete the calendar (member) from the library. Multiple calendars can be selected.

### **Year List Screen**

The screen displays the list of years for which a specified calendar is defined. This screen can be entered directly through the entry panel or the Calendar List screen, or upon returning from the Year Definition screen.

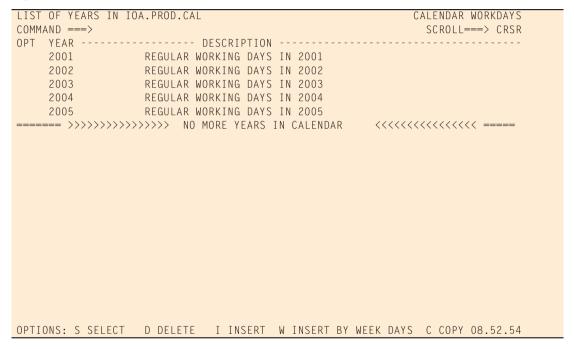
#### - NOTE -



If the S (Select) option was entered in the Calendar List screen for a calendar that is currently in use (selected) by another user, either the Year List screen is not displayed and the Calendar List screen remains displayed (the default), or the Year list screen is displayed in Browse mode (if a user profile definition overrides the default). In either case, an appropriate message is displayed.

If a calendar description was defined in the Calendar Definition screen, the definition is displayed to the right of the year.

Figure 97 Year List Screen



To return to the Calendar List screen press **END** (**PF03/PF15**).

### Format of the Year List Screen

Next to each year in the Year list, certain information can be displayed. The type and format of this information depends on whether the screen is displayed in DESC format or in STAT format:

- In DESC format, the description of the year, taken from the **DESC** field of the calendar definition, is displayed. Default.
- In STAT format, the ISPF statistical information for the calendar definition is displayed.

By default, the Year list is displayed in DESC format. To change formats, use the DESC or STAT commands, described in Table 126.

### **Commands of the Year List Screen**

The following commands can be entered in the **COMMAND** field of the Year List screen.

Table 126 Commands of the Year List Screen

Command	Description
DESC	The DESC command displays the calendar description next to the year. The description is taken from the <b>DESCRIPTION</b> field in the calendar definition.
STAT	The STAT command displays the following ISPF-like statistical information about the calendar next to the year: version and modification numbers, creation date, last modification date, and user ID.

# **Options of the Year List Screen**

To request one of the options shown in Table 127, type the option in the **OPT** field to the left of the year and press **Enter**.

#### NOTE



If the Year List screen is displayed in Browse mode, options D (Delete), I (Insert), and W (Insert By Week Days) are not available.

Table 127 Options of the Year List Screen (part 1 of 2)

Option	Description
S (SELECT)	Display the calendar definition for the specific year.
	Parameters can be edited and updated only if the Calendar Definition screen is not displayed in Browse mode. If the Calendar Definition screen is displayed in Browse mode, the screen can only be browsed and parameters cannot be modified.
D (DELETE)	Delete the calendar definition for the specified year.
I (INSERT)	Insert a new year in the Year List screen and display the Calendar Definition screen with a predefined year definition for editing. The predefined calendar definition is defined with the same dates as the year next to which the I (Insert) request was specified. For more information, see "Inserting a New Year" on page 313.

Table 127 Options of the Year List Screen (part 2 of 2)

Option	Description
W (INSERT BY WEEK DAYS)	Insert a new year in the Year List screen and display the Calendar Definition screen for editing a predefined year definition. The predefined year definition is defined with the same days of the week as the year next to which the W (Insert by Week Days) request was specified. For more information, see "Inserting a New Year."
C (COPY)	Copy the year to another calendar, as described in "Copying Years to Another Calendar" on page 314. Multiple years can be selected.

#### **Inserting a New Year**

All calendar definitions identified in the same Year List usually have the same fixed scheduling pattern. Often, this scheduling pattern is based either on dates within a month or on days of the week within the month. For example:

- Calendar QUARTERLY might always indicate scheduling for the last day of March, June, September and December (that is, a scheduling pattern based on dates).
- Calendar WEEKEND might always indicate scheduling all Saturdays and/or Sundays in each month (that is, a scheduling pattern based on days of the week).

This scheduling pattern also applies to new calendar definitions resulting from the insertion of a new year in the Year List screen.

When a year is inserted in the Year List, the IOA Calendar facility automatically generates a predefined calendar definition for the new year, based on the scheduling pattern of the calendar by which the insert request was specified. This frees the user from having to manually define the new calendar. This automatically generated calendar definition can be displayed and modified.

#### NOTE



The Year list must be kept in ascending order without missing years (for example, 2001, 2002, 2003, 2004, 2005). Each new year must be added at the end of the list.

In calendar definitions, a defined scheduling date is described by both the date (month and day) and the day of the week. Because a particular date falls on a different day of the week in different years, it is necessary to indicate whether the scheduling pattern is based on the date or on the days of the week. This is indicated by the specified insert option.

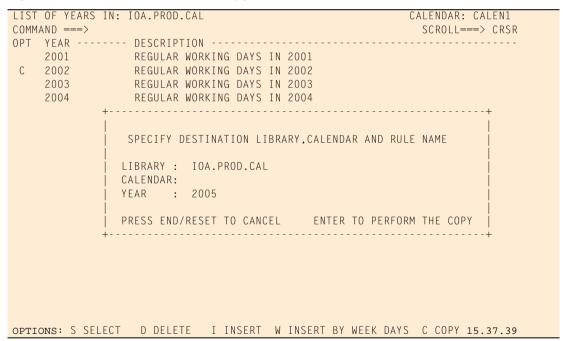
■ To define the calendar with the same scheduling dates (although corresponding days of the week may vary, for example, calendar QUARTERLY described above), type option I (INSERT).

- To define the calendar so that scheduling takes place on the same weekdays as in the previous calendar (although the corresponding dates may vary, for example, calendar WEEKEND described above), type option W (INSERT BY WEEK DAYS).
- If the scheduling pattern is mixed (for example, calendar HOLIDAYS always indicates scheduling on both January 1 and the first Monday in September), specify the more appropriate option and correct the new calendar definition manually.

# **Copying Years to Another Calendar**

Years currently displayed in the Year List screen can be copied to another calendar. To copy the desired years, type option C (COPY) next to each desired year in the screen and press **Enter**. The window shown in Figure 98 is displayed:

Figure 98 Calendar List Screen Copy Window



The window contains the fields shown in Table 128 (some fields contain default values that can be modified):

Table 128 Fields of the Calendar List Screen Copy Window (part 1 of 2)

Field	Description
LIBRARY	Library containing the calendar into which the years must be copied. Must be an existing library. Default: The current library.

Table 128 Fields of the Calendar List Screen Copy Window (part 2 of 2)

Field	Description
CALENDAR	Name of the calendar into which the year must be copied.
	<b>Note:</b> A year can only be copied to another calendar. It cannot be copied to its own calendar (even if the year is renamed).
	If the selected calendar does not exist in the Calendar List, the calendar is created when the request is performed.
YEAR	Name of the year to be copied. If multiple years are selected, the window is initially displayed with the first selected year. As each request is performed or canceled, the next requested year name appears.

To perform a request, press Enter.

To cancel a request, press END (PF03/PF15) or RESET (PF04/PF16).

# **Calendar Definition Screen**

This screen is used to define, display and modify dates in a calendar for a specific year. This screen can be entered directly from the entry panel or from the Year List screen.

Figure 99 Calendar Definition Screen

COMMAND ===>	IOA CALENDAR - WEEKDAYS SCROLL=	(8.Y) ===> CRSR
YEAR 2002	REGULAR WORKDAYS IN 2002	
	SS	S
01 Y Y Y	Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y	
-	SS	S
· -	Y YYYYY YYYY YYYYY YYYY SSS	
1	1 2 3 4 5 6 7 8 9 + 1 2 3 4 5 6 7 8 9 + 1 2 3 4 5 6 7 8	9 + 1
* *	Y YYYYY YYYY YYYYY SSS	•
1 2 3 4 5 04 Y Y Y Y Y	5 6 7 8 9 + 1 2 3 4 5 6 7 8 9 + 1 2 3 4 5 6 7 8 9 + Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y	
	3 4 5 6 7 8 9 + 1 2 3 4 5 6 7 8 9 + 1 2 3 4 5 6 7 8 9 +	
05 Y Y Y	Y	Y
S	1 2 3 4 5 6 7 8 9 + 1 2 3 4 5 6 7 8 9 + 1 2 3 4 5 6 7	-
06	Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y	Υ
-	THE EXECUTION DAYS	14.37.10

### Fields of the Calendar Definition Screen

Table 129 Fields of the Calendar Definition Screen

Field	Description
YEAR	Year of the calendar. This value can be modified. When modified, the values indicated for each date in each month (described below) are shifted to the appropriate day of the week.
description	User-supplied, free text description of the calendar. Optional.
month/dates	Each month of the year (01 through 12) of the calendar consists of the following:
	Separator line. Sunday (or Saturday) is marked "S" (according to the default at your site).
	■ Month label (01 through 12).
	■ Date label for the day of the month.
	Updatable field for defining execution dates. Valid values are:
	■ Y (Yes) – Select the job on that date.
	■ N (No) or ' ' (Blank) – Do not select the job for execution on that date.
	■ + - For a relative calendar, select the closest next "date."
	■ For a relative calendar, select the closest previous "date."
	Note: A relative calendar is a calendar used in a formula to create other calendars. It cannot be specified in a DCAL, WCAL, or CONFCAL field. For details, see the description of the IOABLCAL utility in the INCONTROL for z/OS Utilities Guide.

### **Periodic Calendars**

Some jobs must be scheduled periodically, according to schedules that are not easily expressed in terms of fixed days and dates within months. In these cases, monthly, or even yearly, scheduling definition is awkward. For example:

- A payroll job needs to be scheduled every other Wednesday:
  - In some months, the job may be scheduled on the first, third, and even fifth Wednesday in the month. In other months, it may be scheduled on the second and fourth Wednesday in the month.
  - In some years, the job may be scheduled beginning on the first Wednesday of the year. In other years, it may be scheduled beginning on the second Wednesday of the year.

■ A job must be scheduled every 25 days, regardless of date. Such a job is scheduled on different dates each month and each year.

The IOA Calendar facility provides special calendars, called periodic calendars, to allow specification of these scheduling requirements. These periodic calendars are very flexible.

To designate a calendar as periodic, you must type reserved string ==PERIODIC== in the first 12 positions of the description field. Any text can be entered in the rest of the description field. This is illustrated in Figure 100.

Figure 100 Use of Reserved String "==PERIODIC=="

COMMAND ===>	SCROLL===> CRSR
YEAR 2002 - ==PERIODIC== GENERAL WORKDAY CALENDAR	

The following are characteristics of periodic calendars:

- In a periodic calendar, days are not marked using the letters Y (Yes) or N (No). Instead, a period identifier is used to mark working days in a period. A period identifier can be any letter from A to Z (except Y and N), any number from 0 to 9, or any other printable sign. If you need more characters, use characters falling within the hexadecimal range 4A through F9. All working days within the same period must be marked using the same period identifier character so that different identifier characters indicate different periods. Days that are not marked are nonworking days because they do not belong to any period in this calendar.
- Identifiers from different periods can be interspersed throughout a periodic calendar.
- A periodic calendar can consist of smaller units that do not correspond to regular months, in that they can be longer or shorter than regular months.
- A periodic calendar can span a period, called a "logical year", which can be longer or shorter than one regular calendar year.

When a periodic calendar spans parts of two regular calendar years, special considerations are likely to arise. For more information, see "Special Year-End Handling of Periodic Calendars" on page 319.

■ A period can span any number of days, but no more than a preset number of days can elapse after the appearance of one identifier in a period until the appearance of the next matching identifier in the same period. After that period expires, the next matching identifier starts a new period.

By default, this period is preset to 33 days. Once the length of the gap between matching identifiers exceeds 33 days, the period automatically closes.



#### NOTE -

The length of the default period can be changed from 33 days by the INCONTROL administrator, using optional Wish WM2888.

For more information on the use of periodic calendars, see "DAYS: Basic Scheduling Parameter" on page 420 and "WDAYS: Basic Scheduling Parameter" on page 669.

#### **Examples**

Figure 101 shows examples of periodic calendars:

#### Figure 101 Periodic Calendar – Example 1

This example contains two periods: A and B.

- Period A starts on December 13 and ends on December 23. During this period, the defined working days are December 13, December 18, December 20, and December 23.
- Period B spans more than one calendar year. It starts on December 21 and ends on January 24. During this period, the defined working days are December 21, January 4, and January 24.

Figure 102 Periodic Calendar – Example 2

This example includes a period B that begins on March 9. The last marked working day of the period is March 21, which is followed by a 33-day gap. Assuming that Wish WM2888 has not been used to alter the default period of 33 days, period B automatically ends on April 23, and April 24 marks the beginning of a new period B. If no more B identifiers are added, this new B period ends on May 27.

### **Special Year-End Handling of Periodic Calendars**

Jobs or missions may be improperly scheduled if both the following are true:

- a periodic calendar contains one or more periods that start in one year and continue into the next year
- the first occurrence of the matching identifier in one logical year falls within the default gap that began at the last occurrence of the matching identifier in a prior logical year, possibly as a result of changes made using optional Wish WM2888

In such cases, the period in the prior logical year overlaps the period in the later logical year, causing a scheduled job not to run in the later logical year as expected.

To avoid this problem, remove logical years from periodic calendars as soon as they are no longer needed.

# **Example**

- Logical year FISCAL01 extends from April 1, 2001 through March 31, 2002.
- Logical year FISCAL01 contains a period identified as Period A that has been defined to begin on December 28, 2001 and to continue through January 15, 2002.
- Logical year FISCAL02 extends from April 1, 2002 through March 31, 2003.
- Logical year FISCAL02 also contains a period identified as Period A, defined to begin on April 20, 2002 and continue through May 3, 2002.
- Job X is scheduled for the seventh day of Period A in each logical year, through the job definition DAYS=D7PA.

In a case where the default gap is 33 days, Job X runs in January 2002, and again in April 2002, as expected.

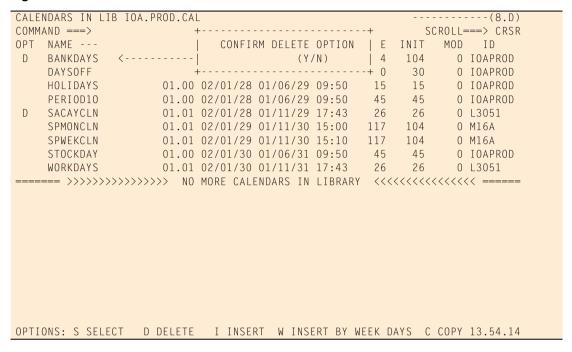
In a case where the default gap is changed from 33 days to a longer period, such as 120 days, the first day of Period A in logical year FISCAL02 occurs less than 120 days after the last appearance of Period A in logical year FISCAL01. As a result, what appears to be the seventh day in Period A in April 2002 is not recognized as such, because the "old" Period A overlaps the "new" Period A. Consequently, Job X does not run again when the user may have expected it to run.

### **Deleting Calendars**

To delete calendars, type option D next to the calendar names in the Calendar List screen and press **Enter**.

The confirmation window shown in Figure 103 is displayed, in sequence, for each calendar selected for deletion.

Figure 103 Calendar List Screen Delete Confirmation Window



Type **Y** (Yes) in the window to delete the calendar.

Type N (No) in the window to cancel the delete request.

#### NOTE -



If PDSMAN is operational at your site, \$\$\$SPACE members are not deleted.

For each calendar deleted, a message is written to the IOA Log file.

### **Exiting the IOA Calendar Facility**

When exiting the IOA Calendar facility, screens are exited in the following sequence:

- 1. Calendar Definition screen
- 2. Year List screen
- 3. Calendar List screen

#### - NOTE -



If the Calendar List screen was bypassed as you entered the IOA Calendar facility (that is, if you entered a CALENDAR value in the entry panel), the Calendar List screen is not displayed upon exiting the Year List screen; instead, the entry panel is displayed.

### **Calendar Facility Entry Panel**

The commands and options available when exiting screens depend on the screen being exited and on whether changes have been made. If changes have been made, the selected exit options and commands determine whether the changes are saved. Exit options and commands are discussed below on a screen by screen basis.

### **Exiting the Calendar Definition Screen**

Use any of the following commands, or press the corresponding PFKey, to exit the Calendar Definition screen:

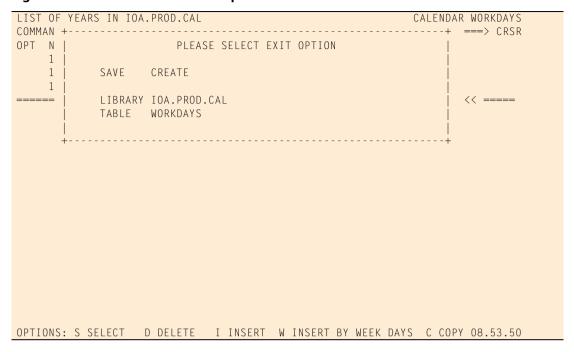
**Table 130 Commands for Exiting the Calendar Definition Screen** 

Command	Description
CANCEL	Cancel the changes made to the calendar definition and return to the Year List screen.
<b>Note:</b> The following exit commands retain changes to the calendar definition in memory. To permanently save the changes to disk, you must also request that the changes be saved when you exit the Year List screen.	
END (PF03/PF15) Enter	Keep changes to the calendar definition in memory and exit to the Year List screen.
NEXTYEAR (PF11/PF23)	Keep changes to the calendar definition in memory and display the next calendar definition in the Year List screen.
PREVYEAR (PF10/PF22)	Keep changes to the calendar definition in memory and display the previous calendar definition in the Year List screen.

### **Exiting the Year List Screen**

Press END (**PF03/PF15**) to exit the Year List screen. If changes made to at least one calendar definition have been kept in memory or if any changes have been made to the Year List screen, the Exit Option window is displayed. For more information, see "Exiting the Calendar Definition Screen" on page 321.

Figure 104 Year List Screen Exit Option Window



Fill in the Exit Option window as follows:

The **LIBRARY** and **TABLE** (member) fields indicate the library and member in which the calendar definitions must be saved. The specified values can be modified (for example, to save the calendar definitions in a different member).

- To save all changes currently in memory and exit the Year List screen, type Y (Yes) after the word SAVE or CREATE:
  - Type Y after the word SAVE if a member with the same calendar name already exists in the specified library.
  - Type Y after the word CREATE if a member with the same calendar name does not exist in the specified library.

#### NOTE -



If you create a new calendar member, the member name does not appear in the Calendar List screen upon exiting the Year List screen; it first appears when you reenter the Calendar List screen from the entry panel.

- To cancel changes currently in memory and exit the Year List screen, type N (No) after the word SAVE or CREATE.
- To close the Exit Option window and remain in the Year List screen (with the changes remaining in memory), press RESET (**PF04/PF16**).

### **Exiting the Calendar List Screen**

Press END (PF03/PF15) to exit the Calendar List screen.

### **Exiting the Entry Panel**

Press END (PF03/PF15) to exit the entry panel.

# **Utilities Under ISPF**

Several IOA facilities can only be activated under ISPF. To activate these facilities, select option 6 on the IOA Primary Option menu (under ISPF) or invoke the IOAUTIL CLIST from the TSO Command Processor. The IOA Online Utilities menu is displayed.

#### - NOTE



The INCONTROL administrator can remove user authority to access option 6 on the IOA Primary Option menu. In this case, the IOA Online Utilities menu is not displayed.

### **IOA Online Utilities Menu**

Depending on the INCONTROL products that are available at your site, different online utility options are displayed in the On-line Utilities menu. Figure 105 shows the IOA On-line Utilities menu that is displayed when all applicable INCONTROL products are active.

Figure 105 IOA Online Utilities Menu when all INCONTROL Products are Installed

```
-----ON-LINE UITILITIES ------
                                                                                                                          USFRID - NO6
                                                                                                                          TIME
                                                                                                                                           - 13:40
                                                                                                                          TERMINAL - 3278
      D1 DECOLLATING
D2 PRINT
                                                        - Schedule a Report Decollating Mission
                                                        - Schedule a Printing Mission
      D2 PRINT
      D3 BACKUP/MIGRATION - Schedule a Backup/Migration Mission
     D4 RESTORE - Schedule a Restore Mission

I1 PREREQ CONDITION - Add/Check/Delete a Prerequisite Condit

M1 JOB ORDER ISSUE - Issue a Job Order

M2 AUTOEDIT SIMUL - Perform an AutoEdit Simulation

M3 SIMUL/TAPE PULL - Prepare Simulation/Tape Pull List Job

M4 PARAM PROMPTING - Parameter Prompting Facilities

M5 QUICK SCHEDULE - Quick Schedule Definition

M6 USER INTERFACE - End-User Job Order Interface

R1 CTM/RESTART SIM - CONTROL-M/Restart Simulation

R2 DATASET CLEANUP - CONTROL-M/Restart Dataset Cleanup

R3 JOB DATASET LIST - Prepare a Job Dataset List

R4 STANDALONE - CONTROL-M/Restart Standalone

T1 CONTROL-M/Tape SIMUL - Simulate CONTROL-M/Tape Rules
             RESTORE
                                                        - Schedule a Restore Mission
                                                        - Add/Check/Delete a Prerequisite Condition
      T1 CONTROL-M/Tape SIMUL - Simulate CONTROL-M/Tape Rules
                                                         - Exit This Menu
OPTION ===>
```

#### NOTE -



If DOCU/TEXT has also been installed at your site, an additional utility, option U1, is displayed in the Online Utilities menu.

To access an available utility, type the desired option number in the OPTION field and press **Enter**.

Options I1, M1, M2, M3, M4, M5, and M6, which are also available when CONTROL-M is installed as a standalone product are described on the following pages. For the descriptions of other utilities on the menu, see the user guides of the relevant products.

Online utility screens utilize standard ISPF profile capabilities.

Quick transfer to a utility can be performed by entering =*opt* from another utility screen, or =6.*opt* from a non-utility screen (for example, the IOA Log screen), where *opt* is the 2-character option identified on the IOA Online Utilities menu.

### 11: Add, Check, or Delete a Prerequisite Condition

This utility adds prerequisite conditions to, checks the existence of prerequisite conditions in, and deletes prerequisite conditions from, the IOA Conditions file.

The Prerequisite Condition Utility screen, shown in Figure 106, can be displayed in the following ways:

- Select Option I1 in the Online Utilities menu.
- Invoke the IOACCND CLIST from the TSO Command Processor screen.

Figure 106 Prerequisite Condition Utility Screen

```
COMMAND ===>

FUNCTION ===> ADD (ADD/CHECK/DELETE)

CONDITION NAME ===> SALARY_RPT_OK

Enter either date or STAT:

CONDITION DATE ===> STAT (DDMM OR STAT)

ENTER YES TO CONTINUE ===> YES
```

To activate the utility, fill in the fields shown in Table 131 and press Enter:

Table 131 Prerequisite Condition Utility Screen Fields (part 1 of 2)

Field	Description
FUNCTION	Function to be performed. Valid values are:
	■ ADD – Add the specified condition to the IOA Conditions file.
	<ul> <li>CHECK – Check if the specified condition exists in the IOA Conditions file.</li> </ul>
	■ DELETE – Delete the specified condition from the IOA Conditions file.
CONDITION	Name of the prerequisite condition (1 through 39 characters) to be
NAME	added, checked, or deleted. If CONDITION NAME values contain the special characters ampersand (&) or apostrophe ('), they must be repeated in order to appear on the screen.

Table 131 Prerequisite Condition Utility Screen Fields (part 2 of 2)

Field	Description
CONDITION DATE	4-character date associated with the specified condition. Valid values are:
	<ul> <li>date – Valid date in date in mmdd or ddmm format, depending on the site standard.</li> </ul>
	<ul> <li>STAT – Static. Value assigned to conditions that are not date-dependent, such as DATABASE-OK.</li> </ul>
ENTER YES TO CONTINUE	Confirmation field to prevent the unintentional addition or deletion of a condition. When blank, the operation is not performed. Type YES to add, check or delete the condition.

To exit the screen without activating the utility, press PF03/PF15.

# M1: Issue a Job Order

This utility is used to issue manual job orders.

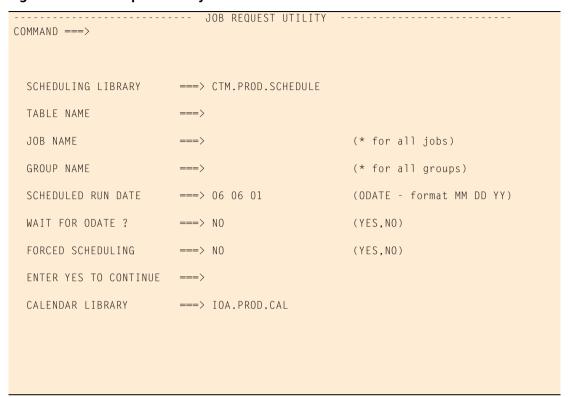
Although most job orders are requested by User Daily jobs that are automatically submitted by CONTROL-M, it is sometimes necessary to issue a job order manually, as in the following situations:

- to order a special purpose job
- to issue a job order for a different working date, for example
  - to reschedule a job run from the 1st of the next month to the 30th of this month
  - to reschedule the run of an entire scheduling table from the 4th of the month to the 5th of the month because all job runs in the table must be performed again

The utility screen (Figure 107) is displayed in the following ways:

- Select option M1 on the Online Utilities menu.
- Activate CLIST CTMJOBRQ from the TSO Command Processor.

Figure 107 Job Request Utility Screen



To activate the utility, fill in the fields shown in Table 132 and press Enter:

Table 132 Parameters of the Job Request Utility Screen (part 1 of 2)

Parameter	Description
SCHEDULING LIBRARY	Name of the scheduling library containing the table or jobs to be scheduled.
TABLE NAME	Name of the scheduling table (member).
JOB NAME	Name of the job to be scheduled.
	If you type an asterisk (*) in this field, all jobs in the specified table are ordered.
GROUP NAME	Name of the group to be scheduled.
	If you type an asterisk (*) in the field, all groups in the specified table are eligible for ordering. The Group name and Job name are logically 'ANDED' to determine whether a job will be selected for scheduling.
SCHEDULED RUN DATE	Original scheduling date of the job or jobs. The default is the current working date.
WAIT FOR ODATE	For a description of the WAIT FOR ODATE parameter, refer to page 153.

Table 132 Parameters of the Job Request Utility Screen (part 2 of 2)

Parameter	Description	
FORCED SCHEDULING	Determines whether the specified job or jobs are forced.	
	Valid values are:	
	<ul> <li>Y (YES) – Schedule the job (or jobs) even if the requested date is not a scheduling date for the job according to its basic scheduling parameters.</li> </ul>	
	<ul> <li>N (NO) – Schedule the job (or jobs) only if the requested date satisfies the basic scheduling criteria of the job.</li> </ul>	
	<b>Note:</b> Jobs in group scheduling tables must be forced. Merely ordering them is not sufficient.	
ENTER YES TO CONTINUE	Confirmation field to help prevent the job (or jobs) from being unintentionally run.	
	Valid values are:	
	<ul> <li>Y (Yes) – The job will run.</li> <li>' (Blank) – The job will not run.</li> </ul>	
CALENDAR LIBRARY	Name of the calendar library (if used) for scheduling the job or jobs. If no calendar library name is specified, the utility uses the calendar library or libraries that are allocated to the DACAL DD name in the online environment.	

To exit the screen without activating the utility, press PF03/PF15.

# M2: Perform an AutoEdit Simulation

This utility checks AutoEdit control statement syntax in jobs. It is essential that the syntax of AutoEdit control statements be checked while the member is being prepared. Otherwise, CONTROL-M may detect an AutoEdit syntax error during job submission in the production environment and cancel the submission.

The utility can be initiated either online (through this screen) or by batch procedure CTMAESIM. For more information, see "Testing AutoEdit Syntax" on page 793.

To activate the utility online, display the utility screen in either of the following ways:

- Select option M2 on the Online Utilities menu
- Activate CLIST CTMCAES from the TSO Command Processor





By default, the utility allocates several files with the TSO user ID as the file name prefix. To change this, activate CLIST from the TSO Command Processor as follows:

TSO CTMCAES PREFX(alternate-prefix)

The CTMCAES utility can operate in either JCL library mode or scheduling library mode:

- In JCL Library mode, the utility checks the AutoEdit statements in the job's JCL.
- In scheduling library mode, the utility not only checks the AutoEdit statements in the job's JCL, it also checks the impact that SET VAR statements in the job scheduling definition have on the job's JCL.

#### NOTE



Started tasks (STCs) are not supported in scheduling library mode.

This facility simulates the actions of the CONTROL-M submission mechanism and produces a printed report of the process.

The output of the simulation process is a standard print file containing

- input control statements
- log messages of the submission process
- actual lines that are submitted under the same conditions

#### NOTE



During AutoEdit simulation, some variables may not contain valid or expected values. For example, %%\$TAG is always blank and %%ORDERID is ZZZZZ.

Figure 108 CONTROL-M AutoEdit Simulation Screen

```
------PERFORM CONTROL-M AUTOEDIT SIMULATION ---------
COMMAND ===>
SPECIFY JCL LIBRARY OR SCHEDULE LIBRARY INFORMATION
JCL LIBRARY MODE:
   JCL LIBRARY
MEMBER NAME
                        ===> CTM.PROD.JCL
                        ===> BRCCIND
   OWNER
                         ===> M21
   APPLICATION NAME
   GROUP NAME
   SCHEDULE TAG NAME
SCHEDULING LIBRARY MODE:
   SCHEDULING LIBRARY
                        ===>
   TABLE NAME
                        ===>
   JOB NAME
                         ===>
PARAMETER LIBRARY ===> CTMP.PROD.PARM
                         ==> 06 06 01 (DD MM YY)
==> 06 06 01 (DD MM YY)
ODATE
FUNCTION
                         ===> LIST
                                         (LIST/SUBSCAN/SUBMIT)
Enter YES to continue ===>
```

The submission simulation utilizes control statements that are written to DD statement DASIM. These control statements are based on the parameters described below.

Depending on the mode in which the utility operates, either JCL library mode or scheduling library mode parameters (but not both) must be specified. In addition, General simulation parameters must also be specified.

To activate the utility, fill in the parameters and press **Enter**.

# **JCL Library Mode Parameters**

**Table 133 JCL Library Mode Parameters** 

Parameter	Description
JCL LIBRARY	Name of the JCL library from which the required JCL is to be "submitted" by the AutoEdit simulation.
MEMBER NAME	Name of the JCL member to be "submitted" by the AutoEdit simulation.
OWNER	User ID of the job's owner.
APPLICATION NAME	Name of the application as specified in field APPL in the job scheduling definition.
GROUP NAME	Name of the group to which the job belongs.

# **Scheduling Library Mode Parameters**

**Table 134 Scheduling Library Mode Parameters** 

Parameter	Description
SCHEDULING LIBRARY	Name of the library containing the job scheduling definition.
TABLE NAME	Name of the scheduling table containing the job scheduling definition.
JOB NAME	Name of the job scheduling definition.

#### NOTE



When specifying scheduling library mode parameters, values for owner, application name, and the JCL library and member of the job are not specified because the utility takes these values directly from the specified job scheduling definition.

The name of the JCL member is obtained from the OVERLIB parameter (if specified) instead of the MEMLIB member.

Table 135 AutoEdit Simulation (part 1 of 2)

Parameter	Description
CONTROL-M GLOBAL LIBRARY	Name of the library that contains the members referenced by AutoEdit statement %%GLOBAL.
WDATE	Working date of the job.
ODATE	Original scheduling date of the job.
FUNCTION	Function to be performed by the simulation. Valid values are:
	■ LIST – The utility simulates submission of the member from the designated library using the specified date and user ID parameters. CONTROL-M checks the JCL. The output is displayed on the terminal. The JCL is not actually submitted.
	■ SUBMIT – CONTROL-M attempts to resolve the AutoEdit statements. If successful, the JCL member lines are also written to the file referenced by the DASUBMIT DD statement and the member is submitted by the utility for execution. In this case, MVS also checks the JCL. This option can also be used to submit jobs when the CONTROL-M monitor is not active (for example, if there is a severe technical problem).
	■ SUBSCAN – This function is similar to SUBMIT except that it adds a TYPRUN=SCAN parameter to the job card before performing simulation. As a result, the job is submitted and the JCL is checked by MVS but the job is not executed.

Table 135 AutoEdit Simulation (part 2 of 2)

Parameter	Description	
	<b>Note:</b> When executing the AutoEdit simulation SUBSCAN function, any JES statements that can cause the console subsystem to be triggered are removed from the job's runstream. Examples of such JES2 statements are:	
	/*MESSAGE /*NETACCT /*NOTIFY /*OUTPUT /*PRIORITY /*ROUTE /*SIGNxxx /*SETUP /*XMIT	
	JES3 statements that have similar effects are also removed.  Utilize this function also if a JCL-checking product (other than JOB/SCAN or PRO/JCL) is in use at the site. The utility creates a copy of the JCL member with all CONTROL-M AutoEdit variables resolved. The JCL-checking product can then be invoked against this copy.	
	■ JOBSCAN – This option is available at sites where the JOB/SCAN or PRO/JCL product is installed, but only if the utility is activated from the Online Utilities menu. (This option is not displayed and cannot be used if the utility is activated by a CLIST or batch procedure.) This function is similar to SUBMIT except that if CONTROL-M finds no JCL errors, JCL is checked by JOB/SCAN before it is written to the file referenced by the DASUBMIT DD statement.	
Enter YES to continu	Confirmation field to help prevent the simulation jobs from being unintentionally run. When blank, the jobs do not run. Enter YES to enable the job run.	

To exit the screen without activating the utility, press PF03/PF15.

# M3: Prepare Simulation/Tape Pull List Job

This screen is used to activate the Simulation procedure or the Tape Pull List procedure. The screen can be displayed in the following ways:

- Select option M3 from the Online Utilities menu
- Activate CLIST CTMCSIM from the TSO Command Processor

Figure 109 CONTROL-M Simulation and Forecasting Facility and Tape Pull List

```
----- CONTROL-M SIMULATION AND FORECASTING FACILITY AND TAPE PULL LIST -----
COMMAND ===>
  RUN SIMULATION
                                              (Y-to run, N-skip to reports)
                      ===> 200106030900 (Format YYYYMMDDhhmm)
     Until
                         ===> 200106031600 (Format YYYYMMDDhhmm)
  ON Today's-current AJF ===> Y
                                              (Y/N If "N", fill in the date)
                                              (DD MM YY)
      Another day - DATE ===>
     Create new AJF ===> N
Order daily jobs ===> N
                                              (Y/N)
      Order daily jobs
                                              (Y/N)
      Keep output AJF, RES ===> Y
                                              (Y/N)
  Parameters member ===> SIMPARM
REPORTS Jobs left ===> Y
                                              (Simulation parameters)
                                              (Y/N)
           Night schedule ===> Y
                                              (Y/N)
  TAPE PULL LIST
                                              (Y/N)
      Report by VOLSER
                         ===> Y
                                              (Y/N)
                                              (Y/N)
      Report by TIME
                         ===> Y
      Report by JOBNAME ===> N
                                              (Y/N)
      Report by DSN
                         ===> N
                                              (Y/N)
      Parameters member ===> TAPULPRM
                                              (Tape pull parameters)
  Enter YES to continue ===> YES
                                           or END key to EXIT
```

The Simulation facility simulates the actions of the CONTROL-M monitor under the conditions specified in the simulation parameters.

Online simulation is performed in the CPU without updating the simulation input files, or without performing any other I/O procedure.

#### NOTE -



At sites supporting the JOB/SCAN, PRO/JCL, or DOCU/TEXT Interface, the lower portion of the Simulation screen is modified to contain the INVOKE JOBSCAN parameters.

The Tape Pull List procedure creates a list of all tapes to be mounted in a specified period, taking into account the expected order of job execution and the order of creation of tape data sets. The list can be sorted and edited in various ways.

This utility also provides the following benefits:

- It checks the syntax of all AutoEdit statements in all jobs that are planned for the given period.
- It checks the JCL syntax.
- It produces a list of data sets that are not available. These are usually input data sets due to arrive, but they may indicate JCL errors. For more information, see the discussion of CTMRNSC, the night schedule report, in the *INCONTROL* for z/OS Utilities Guide.

### – NOTE



For the Tape Pull List procedure to execute properly, authority must be granted for the submission of jobs to the internal reader (INTRDR).

For more information about Simulation and Tape Pull List procedures, see Chapter 7, "Simulation and Forecasting Facility."

To activate this online utility, fill in the fields and sub-fields shown in Table 136, and press **Enter**.

Table 136 Fields of the CONTROL-M Simulation and Forecasting and Tape Pull List Screen (part 1 of 4)

Field	Description
RUN SIMULATION Fields:	
RUN SIMULATION	Whether to run the simulation. Valid values are:
	■ Y (Yes) – Run the simulation. The results of the simulation run are kept in the Log file and the Active Jobs file (AJF) and can be used for producing reports and/or the tape pull list. Default.
	■ N (No) – Do not run the simulation. Use the results of a prior simulation to produce reports and/or the tape pull list.
From	Simulation start date and time, in the format yyyymmddhhmm.
Until	Simulation end date and time, in the format yyyymmddhhmm.
ON Fields:	
ON Today's-current AJF	Whether to use "Today's" data (that is, the data currently in the AJF) as input for the simulation. Valid values are:  ■ Y (Yes) – Use "Today's" data. If you choose this option, BMC Software recommends that you run the simulation after "Today's" jobs have been placed on the AJF using New Day processing. Default.
	<ul> <li>N (No) – Use data from the date specified in the Another Day - DATE field.</li> </ul>
Another day - DATE	Date to use for scheduling or ordering simulation jobs. The format is ddmmyy, mmddyy, or yymmdd, depending on your site standard. A valid date must be entered when not using "Today's" data (that is, if N is entered in the Today's-current AJF field.)

Table 136 Fields of the CONTROL-M Simulation and Forecasting and Tape Pull List Screen (part 2 of 4)

Field	Description
Create new AJF	Whether to allocate a new AJF to contain jobs for the simulation. Valid values are:  Y (Yes) – Allocate a new AJF. This value must be specified when not using the data currently in the AJF, that is, if N is entered in the Today's-current AJF field.
	■ N (No) – Do not allocate a new AJF. This value must be specified when using the data currently in the AJF, that is, if Y is entered in the Today's-current AJF field. Default.
Order daily jobs	Whether to load into the new AJF all the jobs that are scheduled to execute on the specified date. Valid values are:
	<ul> <li>Y (Yes) - Load the jobs into the AJF. A User Daily step is entered into the job. This step schedules all the jobs based on their basic scheduling criteria. It is the user's responsibility to ensure that the Table list for this job is up-to-date.</li> <li>This value must be specified when not using "Today's" data and when creating a new AJF, that is, if N is entered in the Today's-current AJF field and Y is entered in the Create new AJF field.</li> </ul>
	■ N (No) – Do not load the jobs into the AJF. This value is generally specified when using "today's" data or when not creating a new AJF (that is, if Y is entered in the Today's-current AJF field or N is entered in the Create new AJF field.) Default.
Keep output AJF,RES	Specifies whether to save the output AJF, the IOA Conditions file, and the CONTROL-M Resources file (that is, the files as they appear at the end of the simulation.) The output files must be kept if you plan to produce reports, such as a Jobs Left report, based on these files. Valid values are:
	■ Y (Yes) – Keep the output files. Default.
Parameters member	■ N (No) – Do not keep the output files.  Name of the member in the CTM PARM library that contains the simulation parameters. This member may contain parameters such as INTERVAL, ADD COND, and so on.
	Default: SIMPARM
REPORTS Fields:	

Table 136 Fields of the CONTROL-M Simulation and Forecasting and Tape Pull List Screen (part 3 of 4)

Field	Description	
Types of reports to be produced.	Valid values for each report type are:	
<ul> <li>Y – the report type is generated</li> <li>N – the report type is not generated</li> </ul>		
<b>Note:</b> This part of the panel is often site-modified. The following are the default report types:		
Jobs left	This report lists the jobs that did not end OK by the end of the simulation (jobs in status WAIT SCHEDULE, EXECUTING, ENDED NOTOK, and so on). This report is identical to KeyStroke Sample report REPJOBMO in the IOA.KSL library. Default: Y	
Night schedule	This report provides a job execution time summary. For more information, see the discussion of CTMRNSC, the night schedule report, in the <i>INCONTROL for z/OS Utilities Guide</i> . Default: Y	
TAPE PULL LIST Fields:		
TAPE PULL LIST	Specifies whether to run the Tape Pull List procedure. The accompanying "Report by" fields specify whether to generate individual Tape Pull reports. Valid values for this field and the accompanying "Report by" fields are:	
	<ul> <li>Y (Yes) – Generate the report.</li> <li>N (No) – Do not generate the report</li> </ul>	
Report by VOLSER	This report is sorted by volume serial number (this includes all tapes from the tape library). Default: Y	
Report by TIME	This report is sorted by the expected mount time. Default: Y	
Report by JOBNAME	This report is sorted by job name. Default: N	
Report by DSN	This report is sorted by data set name. Default: N.	
Parameters member	Name of the member in the CTM PARM library that contains the Tape Pull parameters. This member may contain parameters such as REPBYVOL, REPBYTIME, or REPBYJOB. Default: TAPULPRM	
INVOKE JOBSCAN Fields:		
INVOKE JOBSCAN	These parameters apply only if the JOB/SCAN, PRO/JCL, or DOCU/TEXT Interface is installed at your site. Valid values for the accompanying fields are Y (Yes) or N (No). Only one Y value can be entered.	
	■ Y (Yes) – JOBSCAN or PRO/JCL is invoked, the validation is performed, and the appropriate report is displayed in the utility output.	
	■ N (No) – The specified validation is ignored.	

Table 136 Fields of the CONTROL-M Simulation and Forecasting and Tape Pull List Screen (part 4 of 4)

Field	Description
JCL Checking	If Y is entered  ■ checks the JCL specified in the member referenced by the DAJCLOUT DD statement for errors  and
	■ checks for adequate DASD disk space allocation.
Errors Only	If Y is entered, checks for JCL errors only.
Space Report	If Y is entered, checks for adequate DASD disk space allocation only.
Enter YES to continue Field:	
Enter YES to continue	When set to blank, the jobs are not generated. This prevents the simulation or tape pull list jobs from being unintentionally generated.  Type YES to enable the job run. The file of the simulation is a tailored to your specifications is displayed in ISPE.
	job as tailored to your specifications is displayed in ISPF EDIT. You can submit it, save it for future use, and so on.  Default: YES

To exit the screen without activating either facility, press PF03/PF15.

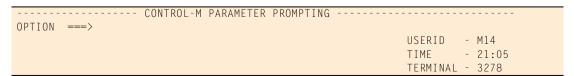
# **M4: Parameter Prompting Facilities**

CONTROL-M Parameter Prompting facilities provide an automatic online interface for assigning values to AutoEdit parameters in the JCL. These facilities prompt the user for values requiring manual modification.

These facilities eliminate the need to remember which AutoEdit parameters require assignment each day, the location of AutoEdit members, and the manual conditions that need to be added (Screen 7).

To display the Parameter Prompting entry panel (below), select option M4 on the Online Utilities menu.

Figure 110 Parameters Prompting Entry Panel



```
1 CTMCFMNU - Parameter Prompting Facility - TYPE 1
2 CTMCAMNU - Parameter Prompting Facility - TYPE 2

X EXIT - Terminate this menu
```

Two different prompting facilities are available:

- Parameter Prompting facility TYPE 1
- Parameter Prompting facility TYPE 2

Using these facilities requires a basic understanding of JCL, the AutoEdit facility, and the concept of prerequisite conditions.

An brief introduction to each of these two types of facilities is presented before the description of the screens in that facility.





For a much more detailed explanation of how the Parameter Prompting facilities work, how they differ from each other, and how to choose the facility that best suits your operational needs, see the discussion of that topic in "Parameter Prompting Facilities" on page 823.

## **Introduction to Parameter Prompting Facility - Type 1**

The Parameter Prompting facility – Type 1 is an ISPF table based facility that provides automatic prompting for AutoEdit parameter values and setting of prerequisite conditions.

This facility is the recommended method for updating AutoEdit parameter members when the parameter value requires manual specification or modification. Frequently, such parameters are associated with prerequisite conditions that must also be manually added to the IOA Conditions file (from the Manual Conditions file).

Example

Tape ABC, which is required by a particular job, arrives from an external location. The volume serial number must specified in the AutoEdit parameter %%ABC_TAPENO, and the condition TAPE_ABC_ARRIVED must be added to the IOA Conditions file, before the job can run.

Without this facility, the user (generally operations personnel) must access the appropriate AutoEdit member and update the parameter value, and must enter the Manual Conditions screen to manually add the required condition to the IOA Conditions file.

With this facility, the user is prompted for the required value. The facility automatically updates the AutoEdit member and adds the related condition to the IOA Conditions file.

Parameter Prompting facility - Type 1 works basically as follows:

1. Using the first option of the facility (DEFINE PARAMETERS AND CREATE A NEW MASTER TABLE), groups of AutoEdit parameters that require value assignment are defined once. These parameters are grouped into a Master Prompting table, the Master table. Default parameter values can be assigned. In addition, prerequisite conditions to be associated with parameters are designated.

The administration of the parameter prompting facility can be decentralized by user groups or applications by choosing a unique CONTROL-M Prompt library on the Primary menu for each application. Master tables defined in a specific Prompt library will be available for update (in option 2) only when that Prompt library is coded on the primary menu. This permits different applications to be concerned only with the master tables associated with that application without cross-interference with other user groups or applications. This design also simplifies security issues.

2. During daily processing, specification of values is made using option 2 (UPDATE PARAMETERS AND SET CONDITIONS). The user selects the desired table from the list of Master tables and is presented with Daily Prompting table - an automatically created copy of the Master table for the current date.

The Daily Prompting table consists of parameter names, (optional) descriptions, and default values. The user updates the desired parameters with the appropriate values.

The facility automatically adds the appropriate conditions to the IOA Conditions file and updates the daily AutoEdit member with the specified values.

# Screens of the Parameter Prompting Facility (Type 1)

After selecting option 1 of the CONTROL-M Parameter Prompting entry panel, the screen shown in Figure 111 is displayed:

### Figure 111 Parameter Prompting Facility (Type 1) Primary Menu

```
OPTION ===>

1 DEFINE PARAMETERS AND CREATE A NEW MASTER TABLE

2 UPDATE PARAMETERS, SET CONDITIONS AND DELETE TABLES

OPTIONS:
PARAMETER DESCRIPTION WILL BE DISPLAYED ===> NO (YES/NO)

IOA CORE PREFIX ===> IOA.PROD

CONTROL-M PROMPT LIB ===> CTM.PROD.PROMPT

CONFIRM PARAMETER UPDATE ACTIONS ===> YES (YES/NO)

ENTER END COMMAND OR PF3 TO TERMINATE
```

#### - NOTE -



You can enter this screen directly by activating CLIST CTMCFMNU.

This screen displays the following options:

1. Define Parameters and Create a New Master Table

This option defines groups of parameters. The definition and association with any prerequisite condition is performed only once per parameter.

2. Update Parameters and Set Conditions

This option is accessed daily (or multiple times in one day) to assign values to parameters and set prerequisite conditions.

The IOA Core Prefix used at your site appears as a default. Files with this prefix are accessed by the Parameter Prompting facility to add prerequisite conditions. Usually, there is no need to change the value of this field.

The library in which the CONTROL-M prompting tables will be placed appears as a default, and can be changed. The ability to decentralize the administration of the parameter prompting facility by using different CONTROL-M prompting libraries is discussed in "Introduction to Parameter Prompting Facility - Type 1" on page 338.

The Confirm Parameters Update Actions field determines whether a confirmation window is displayed following update requests in the Update Parameters and Set Conditions screen, which is described in "Option 2: Update Parameters and Set Conditions" on page 345.

### **Option 1: Define Parameters and Create a New Master Table**

After selecting option 1 of the Parameter Prompting facility (Type 1) Primary menu, the screen shown in Figure 112 is displayed:

Figure 112 Define Parameters and Condition - New Master Table Screen

```
---- CONTROL-M - P.P.F. - DEFINE PARAMETERS AND CONDITIONS -------(P.1)

COMMAND ===>

TABLE NAME PREFIX ===>

Please fill in the TABLE NAME PREFIX and press ENTER

ENTER END COMMAND OR PF3 TO TERMINATE
```

Fill in a Table Name Prefix (a maximum of three characters) and press **Enter**.

A Master table is usually defined for a group of AutoEdit parameters controlled by one person or project.

If the table does not exist (because you are attempting to define a new table), the screen shown in Figure 113 is displayed:

Figure 113 Define Parameters/ and Conditions - Master Table Definition Screen

```
COMMAND ===>
CTMB14E MASTER TABLE TAPTMSTR WAS NOT FOUND. YOU MAY CREATE IT, OR EXIT

TABLE NAME PREFIX ===> TAP

DESCRIPTION ===> EXTERNAL TAPE DATA

Please fill in the Table Description and press ENTER

ENTER END COMMAND OR PF3 TO TERMINATE
```

You can create a new table or exit the screen. To create a new table, enter a table description and press **Enter**.

### **Define Parameters and Conditions Screen**

After creation of a new table, or if the table exists, the following screen is displayed. If the table exists, the previously defined parameters and associated conditions are displayed for modification.

Figure 114 Define Parameters and Conditions Screen

This screen is used to define, display and modify parameters and optional prerequisite conditions that are used for prompting on a daily basis.

## **Specifying Retrieval Criteria**

The display of parameters can be limited to parameters beginning with a specific prefix by filling in the **PARM PREFIX** field.

To display the first occurrence of a parameter at the top of a screen, use the line command L xxxx, where xxxx is a specific parameter or parameter prefix.

### **Define Parameters and Conditions Screen – Format**

The following information can be defined, displayed, or modified for each parameter:

Table 137 Define Parameters and Conditions Screen – Format

Format	Description	
PARM	Name of the AutoEdit parameter.	
CONDITION	Name of a prerequisite condition to be added to the IOA Conditions file when this parameter is updated. Optional.	
VALUE	A default parameter value. Optional.	
DESC	A meaningful description of the parameter.	

### **Define Parameters and Conditions Screen – Options**

To request one of the following options, type the option in the field to the left of the word PARM and press **Enter**.

**Table 138 Define Parameters and Conditions Screen – Options** 

Option	Description	
DELETE	Delete a parameter from the table.	
REPEAT	Duplicate a parameter.	
ADD	Add a parameter (same as option R).	
INSERT	Insert a new parameter in the table. INSERT typed on the Command line inserts a new parameter at the top of the table.	

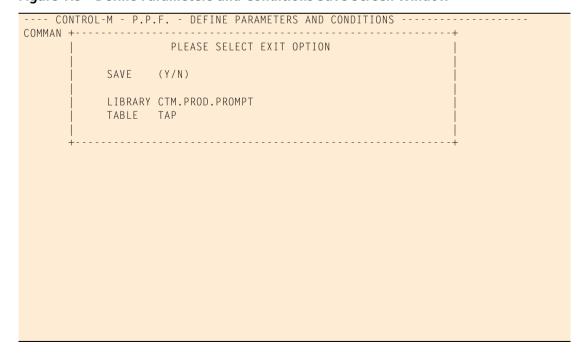
Changes made to a parameter are updated in the plan when you press **Enter**, even if no option is specified.

### **Define Parameters and Conditions Screen – How to Exit**

To exit the Define Parameters and Conditions screen, press END (PF03/PF15).

If additions or modifications have been made, the Save window shown in Figure 115 is displayed:

Figure 115 Define Parameters and Conditions Save Screen Window



Type Y (Yes) to save the changes.

Type N (No) to cancel the changes.

# **Option 2: Update Parameters and Set Conditions**

After selecting option 2 of the Parameter Prompting facility (Type 1) Primary menu, the Table Selection screen is displayed.

Figure 116 Update Parameters and Set conditions - Table Selection Screen

```
---- CONTROL-M - P.P.F. - TABLE SELECTION ------- Row 1 of 3
COMMAND ===>
                                                      SCROLL ===> PAGE
TABLE PREFIX ===>
TABLE NAME ===> BAK
                                               DATE ===> 06 / 06
           LIBRARY : CTM.PROD.PROMPT
           DESCRIPTION: BACKUP CRITERIA
TABLE NAME ===> REP
                                              DATE ===> 06 / 06
           LIBRARY
                     : CTM.PROD.PROMPT
           DESCRIPTION: REPORTING CRITERIA
_ TABLE NAME ===> TAP
                                              DATE ===> 06 / 06
           LIBRARY : CTM.PROD.PROMPT
           DESCRIPTION: EXTERNAL TAPE DATA
***************************** Bottom of Data *******************
```

This screen displays a list of Daily Prompting tables available for update. A Daily table is a copy of a Master Table specific to a particular business day. It is accessed in order to assign values to (previously defined) parameters and to set conditions. The Daily table can be accessed multiple times on the same day.

When you enter this screen, the current date is displayed for each Daily Table. You can overwrite the date to select a different date.

Table deletion can be performed from this screen by typing option D (Delete) in the selection field to the left of TABLE NAME and pressing **Enter**. A Delete Confirmation window is displayed.

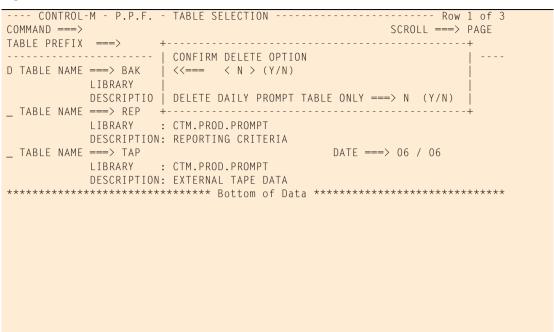


Figure 117 Table Selection Screen Delete Confirmation Window

The Delete Confirmation window also enables you to choose the type of deletion desired.

Type Y (Yes) to confirm the deletion. When deletion is requested, then by default the table is deleted from the Table list, and both the Master Prompting table and the current Daily table are deleted from the Prompting library.

To delete the current daily table without deleting the table from the Table list and without deleting the Master Prompting table from the Prompting library, type Y in the **DELETE DAILY TABLE ONLY** field. In this case, only the current daily table and the corresponding daily AutoEdit member are deleted. The setting of the **DELETE DAILY TABLE ONLY** field is preserved in a profile variable for ease of subsequent use. This is useful if the Master Prompting table has been updated. To reflect those changes in the Daily table and the AutoEdit member, the current Daily table must be deleted, and then be reselected again.

To select a table for any action other than deletion, enter any character except D in the selection field to the left of TABLE NAME and press **Enter**.

The display of tables can be limited to those tables beginning with a prefix of 1 through 3 characters by filling in the **TABLE PREFIX** field. The TABLE PREFIX is retained as an ISPF profile variable from one invocation of the Table Selection screen to the next.

To display the first occurrence of a table at the top of the screen, use the line command L *xxxx*, where *xxxx* is a specific parameter or parameter prefix (under the command line).

## **Update Parameters and Set Conditions Screen**

After table selection, the screen shown in Figure 118 is displayed:

Figure 118 Update Parameters and Set Conditions Screen

This screen displays a list of all AutoEdit parameters for which values can be updated. The following information is presented for each parameter:

**Table 139 Fields of the Update Parameters and Set Conditions Screen** 

Field	Description	
VALUE	Default value of the parameter. This value can be modified.	
OF	Parameter name.	
Description	This description appears only if the value YES was entered in the PARAMETER DESCRIPTION WILL BE DISPLAYED field on the Parameter Prompting facility (Type 1) Primary menu.	
Date Updated	The date of update is displayed in either mm dd or dd mm format depending on the site standard.	

The display of parameters can be limited to parameters beginning with a specific prefix by filling in the PARM PREFIX field (under the command line).

To display the first occurrence of a parameter at the top of a screen, use the line command L parm, where parm is a specific parameter or parameter prefix.

From this screen, conditions can be added to the IOA Conditions file, with or without changing the value of the parameter.

- To add the condition without changing the parameter value, enter any character in the selection field to the left of the VALUE field.
- To update a parameter value and add the condition, update the value as desired and press **Enter**.

If Y (Yes) was entered for the "Confirm Parameter Update Actions" option in the Type 1 Primary menu, the following confirmation window is displayed. Type Y to confirm the updates (or N to cancel them).

Figure 119 Update Parameters and Set Conditions - Confirm Parameter Update Actions

After an update request is completed in the screen, all changes are immediately saved in the Daily table. Any manual condition associated with this parameter prompt is added to the IOA Conditions file.

Press END (PF03/PF15) to exit the screen.

# **Introduction to Parameter Prompting Facility - Type 2**

The Parameter Prompting facility – Type 2 provides automatic prompting for AutoEdit parameter values for manually scheduled jobs. This may be very useful in a distributed environment where user departments are responsible for manually ordering jobs and specifying required parameters.

On any given day, the user selects scheduling tables for execution. The user is then prompted for parameter values required for the execution of those jobs.

The Parameter Prompting facility – Type 2 works as follows:

■ First, relevant scheduling tables are defined or placed in the Master Scheduling Tables library.

Then, using the CREATE AND UPDATE A MASTER PLAN option of the facility, the user defines a Master Prompting Plan (MPP) for each scheduling table in the library. The MMP is placed in the Master Prompting Plan library. It contains all AutoEdit variables used by all jobs in the scheduling table. Default values and value validity checks can also be defined.

Once all definitions are complete, the facility is ready for use on any given day, as needed.

- The user uses the second option of the facility, FETCH A PLAN, to select a plan for execution by CONTROL-M on any specific day.
  - When a FETCH option is executed for a specific plan (or set of plans), a Daily scheduling table is automatically created and placed in the Daily Scheduling Tables library. The Daily Scheduling table is a subset of the Master Scheduling table, and contains the job scheduling definition of each job in the Master Scheduling table scheduled that day.
  - The FETCH also creates a User Prompting Plan (UPP), which is placed in the Daily Prompting Plan library. The UPP is a subset of the Master Prompting Plan, and contains only parameters that are required by the jobs scheduled to run on that day.
  - A Daily JCL library is also created containing JCL for the day's jobs.
- Using the third option of the facility, EXEC A PLAN, the user may then accept or update the default values of all the parameters appearing in the daily UPP.

A daily AutoEdit parameters member, which is accessed at the time of job submission, is automatically created and placed in the Daily AutoEdit Parameter library.

### Screens of Parameter Prompting Facility (Type 2)

After selecting option 2 of the CONTROL-M Parameter Prompting entry panel, the following menu is displayed:

### Figure 120 Parameter Prompting Facility (Type 2) Primary Menu

```
---- CONTROL-M - PARAMETER PROMPTING FACILITY (TYPE 2) PRIMARY MENU ------(P)

OPTION ===>

1 CREATE AND UPDATE A MASTER PLAN

2 FETCH A PLAN (CTMFETCH)

3 EXEC A PLAN (CTMEXEC)

ENTER END COMMAND OR PF3 TO TERMINATE
```

#### - NOTE -



You can enter this screen directly by activating CLIST CTMCAMNU.

This screen displays three options:

1. CREATE AND UPDATE A MASTER PLAN

This option defines groups of parameters in a Master Prompting Plan.

2. FETCH A PLAN (CTMFETCH)

This option places a User Prompting Plan (a copy of the Master Prompting Plan) and related job scheduling definitions in Daily libraries. A "fetch" is required before assigning parameter values and ordering plan execution with Option 3.

3. EXEC A PLAN (CTMEXEC)

This option assigns values to parameters and orders a Plan for execution.

## **Option 1: Create and Update a Master Plan**

After selecting Option 1 of the Parameter Prompting facility (Type 2) Primary menu, the following screen is displayed:

Figure 121 Primary Prompting Facility – Define or Update a Master Plan

```
COMMAND ===>

PLAN NAME IS:

PLAN NAME PREFIX ===> REPTS

LIBRARY ===> CTM.PROD.PLANMSTR

Please fill in the Plan Name Prefix and press ENTER

ENTER END COMMAND OR PF3 TO TERMINATE
```

A Master Plan is usually defined for a group of jobs and their AutoEdit parameters that are controlled by one person or project.

Type a maximum of six characters in PLAN NAME PREFIX and press **Enter**.

The name of the default library in which the Master Plan is placed is displayed. It may be changed.

If the plan does not exist (because you are defining a new plan), the following screen is displayed:

### Figure 122 Parameter Prompting Facility – Master Plan Definition

```
COMMAND ===>

COMMAND ===>

CTMB14E MASTER PLAN REPTS NOT FOUND. YOU MAY CREATE IT, OR EXIT

PLAN PREFIX NAME ===> REPTS

DESCRIPTION ===> DAILY REPORTS

LIBRARY ===> CTM.PROD.PLANMSTR

Please fill in the Plan Description and press ENTER

ENTER END COMMAND OR PF3 TO TERMINATE
```

You can create a new plan or exit the screen. To create a new plan, enter a plan description and press **Enter**.

### **Define Parameters in the Master Plan**

After creation of a new plan, or if the requested plan exists, the following screen is displayed. If the plan exists, the previously defined parameters are displayed for modification.

Figure 123 Define Parameters in the Master Plan Screen

```
---- CONTROL-M - P.P.F. - DEFINE PARAMETERS IN THE MASTER PLAN --- ROW 1 OF 3
COMMAND ===>
                                                       SCROLL ===> CSR
PARM PRFFIX ===>
                                              PLAN NAME: REPTS
PARM NAME ===> REPT NAME
                                              OCCUR NO. ===> 01
   JOB NAME ===> SLSREPTS
                                              PROMPT IND ===> Y (Y/N)
   DEFAULT ===>
   TYPE ===> NONBLANK, MAXL 8
   MESSAGE ===> Enter name of sales report required
_ PARM NAME ===> DEPT_NUMBER
                                              OCCUR NO. ===>
   JOB NAME ===> ******
                                              PROMPT IND ===> Y (Y/N)
   DEFAULT ===> 035
          ===> NUM, MAXL 3
   TYPE
   MESSAGE ===> Enter department number (used for all reports)
_ PARM NAME ===> REPT_NAME
                                              OCCUR NO. ===> 02
   JOB NAME ===> EXPREPTS
                                              PROMPT IND ===> Y (Y/N)
   DEFAULT ===>
   TYPE ===> NONBLANK, MAXL 8
   MESSAGE ===> Enter name of expense report required
******************************* Bottom of Data **************************
```

This screen is used to define, display and modify parameters that are used for prompting on a daily basis.

### Define Parameters in the Master Plan Screen – Format

The following information can be defined, displayed, or modified for each parameter:

Table 140 Fields of the Define Parameters in the Master Plan Screen (part 1 of 2)

Field	Description	
PARM NAME	Name of the AutoEdit parameter.	
OCCUR NO.	Occurrence number (2 digits). Differentiates between use of the same parameter name for different purposes in different jobs (for example, assign OCCUR NO. 01 to occurrence of %%PARM1 in Job A; assign OCCUR NO. 02 to occurrence of %%PARM1 in Job B).	
JOB NAME	Name of the job using the parameter.  If the parameter and its assigned value are shared by more than one job in the plan, enter ******* in this field. It is not necessary to redefine the parameter. (If the value assigned is different for each job, refer to the OCCUR NO. parameter above.)	
PROMPT IND	<ul> <li>Prompting Indicator:</li> <li>■ Y (Yes) – Promptable. The user is prompted for a value for this parameter.</li> <li>■ N (No) – Non-promptable. The value is fixed in the Master Prompting Plan and is not modifiable in the EXEC phase.</li> </ul>	

Table 140 Fields of the Define Parameters in the Master Plan Screen (part 2 of 2)

Field	Description		
DEFAULT	Default value for the parameter that is displayed during the EXEC phase.		
	This field is mandatory if PROMPT IND is set to N (non-promptable).		
	BLANK - Type the word BLANK to set a value of " ".		
ТҮРЕ	Type of parameter value that can be entered. A validation check is performed during both the plan definition and EXEC phases.		
	Valid types:		
	<ul> <li>NUM – Limits the value to digits only (0 through 9).</li> <li>ALPHA – Limits the value to letters only (a-z, A-Z, and \$,#,@).</li> <li>CHAR – Alphanumeric.</li> <li>BLANK – Field must be blank.</li> <li>NONBLANK – Any non-blank value.</li> <li>MINL <i>n</i> – Limits the value to a specified minimum character length, where <i>n</i> is any number from 1 through 70.</li> <li>MAXL <i>n</i> – Limits the value to a specified maximum character length, where <i>n</i> is any number between 1 and 70.</li> <li>MINL, MAXL, and NONBLANK can be combined with NUM or</li> </ul>		
	ALPHA.		
	Example: NUM MAXL 8 limits the parameter value to a numeric value with a maximum length of 8 characters.		
MESSAGE	Prompting message to be displayed during the EXEC phase.		

The display of parameters can be limited to parameters beginning with a specific prefix by filling in the PARM PREFIX field (under the command line).

To display the first occurrence of a parameter at the top of a screen, use the line command L xxxx, where xxxx is a specific parameter or parameter prefix.

### **Define Parameters in the Master Plan Screen – Options**

To request one of the following options, type the option in the field to the left of the words PARM NAME and press **Enter**.

Table 141 Options of the Define Parameters in the Master Plan Screen (part 1 of 2)

Option	Description
D (DELETE)	Delete a parameter from the plan.
R (REPEAT)	Duplicate a parameter.

Table 141 Options of the Define Parameters in the Master Plan Screen (part 2 of 2)

Option	Description
A (ADD)	Add a parameter (same as option R).
I (INSERT)	Insert a new parameter in the plan. INSERT typed on the Command line inserts a new parameter at the top of the plan.

Changes made to a parameter are updated in the plan when you press **Enter**, even if no option is specified.)

#### Define Parameters in the Master Plan Screen – How to Exit

To exit the Define Parameters in the Master Plan screen, type one of the following commands on the command line:

Table 142 Define Parameters in the Master Plan Screen - Exit Screen Commands

Command	Description	
END	Keep all plan changes, and exit.	
CANCEL	Exit without saving plan changes.	

### **Option 2: Fetch a Plan (CTMFETCH)**

After selecting option 2 of the Parameter Prompting facility (Type 2) Primary menu, the following screen is displayed:

Figure 124 Fetch a Plan Screen

```
---- CONTROL-M - P.P.F. ----- FETCH A PLAN -----(P.2)
COMMAND ===>
   PLAN NAME
                        ===> REPTS
   PLAN NAME SUFFIX
                        ===>
                                       (For multiple plans in the same day)
   OVERRIDE DAILY PLAN
                        ===> NO
                                       (YES / NO)
   ODATE
                        ===> 080800
 Please fill in the Plan Name and press ENTER
   MASTER SCHEDULING LIB ===> CTM.PROD.SCHEDULE
   DAILY SCHEDULING LIB ===> CTM.PROD.SCHD
   MASTER PLANS LIB
                       ===> CTM.PROD.PLANMSTR
   DAILY PROMPT PLANS LIB ===> CTM.PROD.PLAN
   MASTER JCL LIB ===> CTM.PROD.JCLPROMP
                        ===> CTM.PROD.JCLP
   DAILY JCL LIB
 ENTER END COMMAND OR PF3 TO TERMINATE
```

This screen places a daily User Prompting Plan (a copy of the Master Prompting Plan) and related job scheduling definitions in Daily libraries. Fill in the details in the screen (libraries and the current date appear as defaults) and press **Enter**.

The PLAN NAME is the same as the Master Prompting Plan PREFIX.

You can designate two characters to serve as a suffix to the Plan Name. This permits execution of a specific plan more than once a day.

Valid values for OVERRIDE DAILY PLAN:

Table 143 Fetch Plan Screen OVERRIDE DAILY PLAN Values

Value	Description
YES	A duplicate fetch of a plan (with a suffix, if one has been designated) replaces an existing copy of a plan with the same PLAN NAME (and same suffix) for that day.
NO	Multiple fetches of a plan are not permitted on the same day.  Default.

## **Option 3: Exec / Order a Plan (CTMEXEC)**

After selecting option 3 of the Parameter Prompting facility (Type 2) Primary menu, the following screen is displayed:

Figure 125 Exec/Order a Plan (CTMEXEC) Screen

```
---- CONTROL-M - P.P.F. ---- EXEC / ORDER A PLAN ------(P.3)
COMMAND ===>
   PLAN NAME
                         ===> REPTS
                                        (Blank for plan selection list)
   PLAN NAME SUFFIX
                                        (For multiple plans in the same day)
                         ===>
   REMAINING PARAMETERS
                         ===> NO
                                        (YES / NO)
   ODATE
                         ===> 080800
   FORCED FROM TIME
                         ===>
 Please fill in the Plan Name (or blanks) and press ENTER
   DAILY SCHEDULING LIB ===> CTM.PROD.SCHD
   USER PROMPT PLANS LIB ===> CTM.PROD.PLAN
   DAILY PARAMETERS LIB ===> CTM.PROD.AEDI
 ENTER END COMMAND OR PF3 TO TERMINATE
```

This screen orders a plan for parameter updating and plan execution. Fill in the details in the screen (libraries and the current date appear as defaults) and press **Enter**.

The PLAN NAME is the same as the Master Prompting Plan PREFIX. You can designate two characters to serve as a suffix to the PLAN NAME. This permits execution of a specific plan more than once a day.

The REMAINING PARAMETERS field determines whether you are automatically prompted in the Update Parameter Values screen for parameter values that have yet to be updated for active plans. Valid values are:

- YES Prompt
- NO Do not prompt

The ODATE field specifies the original scheduling date for executing the plan.

The FORCED FROM TIME field specifies a time (format hhmm) before which the jobs cannot run.

If you leave PLAN NAME blank on the Exec / Order a Plan screen, the Plan Selection screen is displayed:

Figure 126 Plan Selection Screen

This screen displays a list of active Daily Plans.

PLAN ORDERED ALREADY: indicates whether the plan was already ordered. If the plan has already been ordered, it is possible to select a plan for parameter value updating only.

To select a plan, enter any character in the field to the left of the PLAN NAME.

### **Update Parameter Values Screen**

After selecting a plan from the Plan Selection screen or specifying a particular plan on the Exec / Order a Plan screen, the Update Parameter Values screen is displayed:

Figure 127 Update Parameters Values Screen

```
---- CONTROL-M - P.P.F. - UPDATE PARAMETER VALUES ------ (P.3.1)
COMMAND ===>
                                                  SCROLL ===> CSR
                                       PLAN NAME: REPTS
PARM PREFIX ===>
 PARM NAME ===> REPT NAME
                                        OCCUR NO. ===> 01 NO DEFAULT
VALUE ===>
           Enter name of sales report required
 PARM NAME ===> DEPT_NUMBER
                                        OCCUR NO. ===> DEF EXISTS
VALUE ===> 035
           Enter department number (used for all reports)
                                        OCCUR NO. ===> 02 NO DEFAULT
 PARM NAME ===> REPT NAME
VALUE ===>
          Enter name of expense report required
 ************************ Bottom of Data *******************
```

This screen displays a list of all AutoEdit parameters for which values can be entered. Press END (PF03/PF15) to exit the screen.

The Master Prompting Plan for the PLAN NAME is copied from the Master Prompting Library to the Daily Prompting Library to create or replace the corresponding User Prompting Plan (UPP). Only parameters that belong to jobs that meet both of the following criteria are copied into the UPP.

- The job names, which are specified in the DEFINE PARAMETERS IN THE MASTER PLAN screen (option 1), reference a job that appears in the Daily Scheduling table.
- The jobs must be scheduled to run on the specified day (today). The CTMJOB utility is invoked to determine which jobs run today.

The display of parameters can be limited to plans beginning with a specific prefix using the PARM PREFIX field (under the command line).

To display the first occurrence of a parameter at the top of the screen, type the line command L *xxxx*, where *xxxx* is a specific parameter or parameter prefix.

After all variables in a plan have been updated or have had their defaults approved, you receive screen messages indicating the jobs from each plan that were ordered automatically. To cancel any parameter value changes made and return to the Exec / Order a Plan screen, type the line command CANcel.

**Table 144** Format of the Update Parameter Values Screen

Field	Description		
PARM PREFIX	Plan prefix. If a value is entered in this field, the display of parameters is limited to plans beginning with the specified prefix.		
PLAN NAME	Name of the User Prompting Plan ordered for execution.		
PARM NAME	Name of the parameter available for update.		
VALUE	Default value of the parameter. This value can be modified; embedded blanks are permitted.		
MESSAGE	Prompting message.		
OCCUR NO.	Occurrence number (2 digits). Differentiates between use of the same parameter name for different purposes in different jobs (for example, assign OCCUR NO. 01 to occurrence of %%PARM1 in Job A; assign OCCUR NO. 02 to occurrence of %%PARM1 in Job B).		
DEFAULT STATUS	Indication of default:  ■ NO DEFAULT – No associated default value.		
	■ DEF EXISTS – Parameter has an associated default value that has not yet been approved by the user.		
	■ DEF CONFIRMED – Default value has been approved.		
	■ DEF CHANGED- Default value is not being used. Parameter has been assigned a different value.		
SELECTION FIELD	Type S in this field (A) to accept the default, if a default exists.		

## **Special Options**

A special option, activated by typing YES in the REMAINING PARAMETERS field on the Exec / Order a Plan screen, prompts you automatically for parameter values that have yet to be updated from all active plans (that is, those plans "fetched" for the day). The parameters are presented on consecutive Update Parameter Values screens.

- YES You are presented with remaining (non-updated) parameters from active plans.
- NO After updating the current plan, the Exec / Order a Plan screen is displayed or, if Plan Name was left blank, the Plan Selection screen containing all active plans is displayed. Default.

# **M5: Quick Schedule Definition**

The Quick Schedule Definition facility is an automatic online interface for creating scheduling tables for regular jobs that have common scheduling parameters. (Group tables are not supported.) This facility speeds up the process of defining a schedule by eliminating the need to individually define parameters for each job and its job interdependencies.

Twenty-one jobs and their interdependencies can be defined on one screen with CONTROL-M automatically providing space for additional jobs.

The utility can be requested in the following ways:

- Select option M5 on the Online Utilities menu
- Activate CLIST CTMQUICK from the TSO Command Processor

# **Quick Schedule Definition Process**

Four simple steps are performed one time only in order to create a complete scheduling table for an unlimited number of jobs.

Table 145	<b>Ouick Sch</b>	edule Defin	ition Process
-----------	------------------	-------------	---------------

No.	Step	Where Performed
1.	Create a skeleton job.	Screen 2, Scheduling Definition facility.
2.	Specify general table information and prerequisite conditions format.	Quick Schedule Definition entry panel.
3.	List job interdependencies.	Quick Definition Job List screen.
4.	Exit the Quick Schedule Definition facility.	<b>Note:</b> The scheduling table is automatically created upon exit from the Quick Schedule Definition facility.

These steps are described in detail below.

### Step 1: Create a Skeleton Job

In this step you create a job in a scheduling table to be used as a skeleton, or model, for all the jobs in the automatically created scheduling table (output table).

Enter the CONTROL-M Scheduling Definition facility and create a standard CONTROL-M scheduling table containing one skeleton job. For more information, see "Scheduling Definition Facility" on page 110.

Specify in the skeleton job all parameter values that are to be common to (the same in) all the jobs in the automatically created table.

IN and OUT prerequisite conditions are automatically created by CONTROL-M in the output scheduling table; therefore, IN and OUT parameters in the skeleton definition should not be coded, as they will be ignored.

MEMNAME, MEMLIB, and DOCLIB fields are overridden by CONTROL-M during automatic table creation.

The data in all other fields is copied into each of the new jobs in the output table. Therefore, it is important to verify the data carefully.

### %%JOBNAM and %%JOBNAME Variables

If variable %%JOBNAM, a non-AutoEdit variable specific to the Quick Schedule Definition facility, is specified in a SHOUT statement, it is resolved during table creation to the member name in each job.

If System variable %%JOBNAME is specified in a SHOUT statement, it is resolved at runtime to the name of the job. If the job name is not known, %%\$MEMNAME can be used in its place.

### **Step 2: Specify General Table Information and Prerequisite Conditions Format**

In this step, you display the Quick Schedule Definition entry panel and specify general table information and the desired format for automatically defined prerequisite conditions.

The entry panel can be displayed either by requesting option M5 on the Online Utilities menu, or by activating CLIST CTMQUICK from the TSO Command Processor. The following screen is displayed:

Figure 128 CONTROL-M Quick Schedule Definition Screen

```
------ CONTROL-M OUICK SCHEDULE DEFINITION ------
COMMAND ===>
SPECIFY LIBRARY, OUTPUT SCHEDULING TABLE, SKELETON SCHEDULING TABLE
  LIBRARY ===> CTM.PROD.SCHEDULE
  TABLE ===> PAYROLL
                                     (Scheduling table to be created)
  SKELETON ===> DAILY
                                    (Skeleton scheduling table)
  OWNER in the output table S
                                    (T: your TSO User ID)
                                    (S: OWNER from the skeleton table)
PREREQUISITE CONDITIONS FORMAT (CHOOSE ONE)
  GROUP-FROMJOB-SUFFIX ===> Y
                                    (Y/N)
  FROMJOB-TOJOB-SUFFIX ===> N
                                    (Y/N)
  PREFIX-FROMJOB-TOJOB ===> N
                                    (Y/N)
  TOJOB-FROMJOB-SUFFIX ===>
                                    (Y/N)
      PREFIX OR SUFFIX ===> OK
      GROUP ===> FINANCE SERVICES
                                   (For group-fromjob-suffix option)
      CONNECTOR CHARACTER ==>
```

Fill in the following general table information fields:

Table 146 Fields of the CONTROL-M Quick Schedule Definition Screen

Field	Description
LIBRARY	Name of the library that contains the skeleton member created in Step 1 and that will contain the output scheduling table.
TABLE	Name of the scheduling table to be created.
SKELETON	Member name of the model scheduling table containing common parameter values (created in Step 1 above). The member must exist in the library specified above.
OWNER	Value to be entered in the OWNER field in the output scheduling definitions. Valid values are:
	■ T – Your TSO user ID is used as the value for OWNER in the output tables.
	<ul> <li>S – The value of OWNER in the skeleton table is used for OWNER in the output tables.</li> </ul>

To exit this screen, press END (PF03/PF15).





If you use the selection list fields, their values are not erased until you exit the entry panel by pressing END (PF03/PF15).

### **Prerequisite Condition Format Fields**

Job dependencies are established by prerequisite conditions that are defined in the job scheduling definitions.

The utility defines prerequisite conditions automatically. Therefore, naming conventions for these conditions must be specified. Prerequisite conditions created by the utility must consist of a combination of the following elements:

**Table 147 Prerequisite Condition Format Fields** 

Field	Description
FROMJOB	Name of the predecessor job in the dependency.
	For example, if JOB-A must terminate before JOB-B can be submitted, JOB-A is the FROMJOB.
TOJOB	Name of the successor job in the dependency. For example, if JOB-B must be submitted after JOB-A terminates, JOB-B is the TOJOB.
GROUPNAME	Name of the group to which the jobs in the dependency belong.
PREFIX	Constant to be added as a prefix to the condition.
SUFFIX	Constant to be added as a suffix to the condition.

#### - NOTE



Job dependencies are defined in Step 3, described in "Step 3: Specify Job Interdependencies" on page 365.

CONTROL-M can create prerequisite conditions based on the above elements in several different formats. These formats are described below. Select one of the formats by typing Y (Yes) to the right of one desired format, and N (No), to the right of the remaining formats. IN and OUT prerequisite conditions are automatically created in the job scheduling definitions in the selected format.

**Table 148 Formats for Prerequisite Conditions (part 1 of 2)** 

Format	Description
GROUPNAME-FROMJOB -SUFFIX	If Y is entered, creates conditions in the following format (for example): BACKUP-BKP00010-OK.
FROMJOB-TOJOB-SUFFIX	If Y is entered, creates conditions in the following format (for example): BKP00010-BKP00020-OK.

Table 148 Formats for Prerequisite Conditions (part 2 of 2)

Format	Description
PREFIX-FROMJOB-TOJOB	If Y is entered, creates conditions in the following format (for example): VALCHK-BKP00010-BKP00020.
TOJOB-FROMJOB-SUFFIX	If Y is entered, creates conditions in the following format (for example): BKP00020-BKP00010-OK

The following fields affect the above formatted conditions. The GROUP field also affects the GROUP value in the job scheduling definition.

**Table 149 Fields that Affect Prerequisite Conditions Formats** 

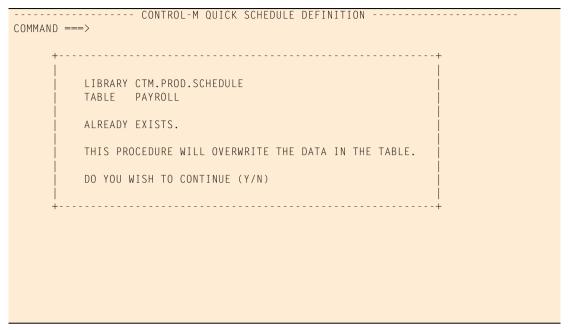
Field	Description
PREFIX OR SUFFIX	Constant to be used as a prerequisite condition prefix or suffix (depending on the format selected). Mandatory. Valid values are: 1 through 9 characters.
GROUP	1 through 20 character group name (no embedded spaces) to be used in the job scheduling definitions. Optional, except for format GROUP-FROMJOB-SUFFIX (for which it is mandatory).  If specified, the value in this field is used as the GROUP value in the created job scheduling definitions (that is, in place of the GROUP value in the skeleton).
	If the GROUP-FROMJOB-SUFFIX format is requested, an * (Asterisk) can be entered in this field. In this case, the group name is omitted from the prerequisite condition (such as BKP00010-OK), but the created job scheduling definitions still contain the group name defined in the skeleton.
CONNECTOR CHARACTER	Character used to concatenate the components of the condition names. Mandatory. Valid values are: one non-blank character other than '&' (Ampersand), for example, '-'.

### **Proceeding to the Job List Screen**

Once you have filled in the fields in the Quick Definition entry panel, press Enter.

- If the table that you specified in the TABLE field does not already exist in the library, the Job List screen is displayed and you can proceed with Step 3.
- If the table that you specified in the TABLE field already exists in the library, the Overwrite Confirmation window is displayed:

Figure 129 CONTROL-M Quick Search Schedule Definition



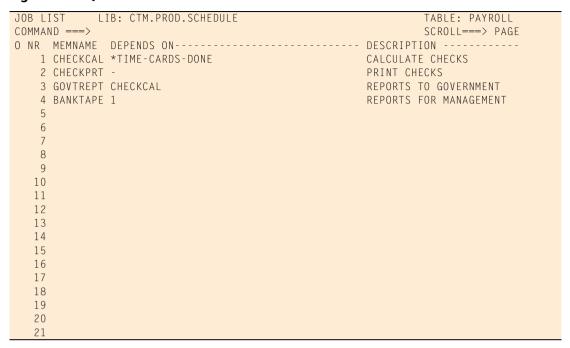
- Type Y (Yes) to overwrite the existing table. The current contents of the table are erased, and an empty table (Job List screen) is displayed.
- Type N (No) if you do not want to overwrite the current contents of the table. The window is closed. You can now type a different table name in the TABLE field and press **Enter** again.

### **Step 3: Specify Job Interdependencies**

In this step you fill in a list of jobs, a description of each job, and the jobs upon which they depend.

After you fill in the Quick Schedule Definition entry panel (and, if necessary, the Overwrite Confirmation window) and press **Enter**, the Job List screen is displayed:

Figure 130 Quick Schedule Definition Job List Screen Entered



Fill in one line for each job (the fields are detailed below). CONTROL-M provides additional lines on the screen, as necessary. When you have finished filling in the list, press **Enter**. The entries are validated.

### Fields in the Job List Screen

Table 150 Fields in the Job List Screen (part 1 of 2)

Field	Description
0	Field for specifying options, which are described in Table 151 on page 367.
NR	Line number. This number can be referenced in the DEPENDS ON field of another job.
MEMNAME	Name of the member containing the JCL of the job.  Names that have DUMMY as a prefix cause the utility to create the job scheduling definition as a dummy job.

Table 150 Fields in the Job List Screen (part 2 of 2)

Field	Description
DEPENDS ON	Jobs and/or external prerequisite conditions on which this job depends. Valid formats for the dependencies:
	■ <i>name</i> – Name of the job (MEMNAME) upon which the current job depends.
	<ul> <li>position-number – Number of the job on the screen. This number is automatically adjusted when an option changes the position of the current job or the job upon which it depends.</li> </ul>
	<ul> <li>- (Minus sign) – The Minus sign represents the previous job in the list.</li> </ul>
	*in-condition – Name of an external prerequisite condition, that is, a prerequisite condition other than job interdependencies that are automatically created. It must be preceded by an asterisk (*) and be the last dependency entered on the job line. The date reference ODAT is automatically associated with the in-condition.
	More than one dependency can be listed by separating each name by a comma. Format types may be mixed on a line.
	Examples:
	<ul> <li>CHECKCAL – Job CHECKCAL</li> <li>1 – Job on line 1 of the list</li> <li>(Minus sign) – Job on the preceding line</li> <li>3,*SALES-DATA – Job on line 3 of the list plus an external IN condition.</li> </ul>
DESCRIPTION	Description of the job in free text.

### **Options of the Job List Screen**

To use one of the following options, type the option in the O field to the left of the line number. These options are similar to ISPF line commands.

Table 151 Options of the Job List Screen (part 1 of 2)

Option	Description
I	Insert a blank line immediately after the current line.
P	Insert a blank line immediately preceding this line. This enables addition of data before the first line in the list.
R	Repeat this line immediately after the current line.
D	Delete this line. If a job depends upon this line, you receive an error message.
A	Indicates that the target of a copy or move is directly after this line.

Table 151 Options of the Job List Screen (part 2 of 2)

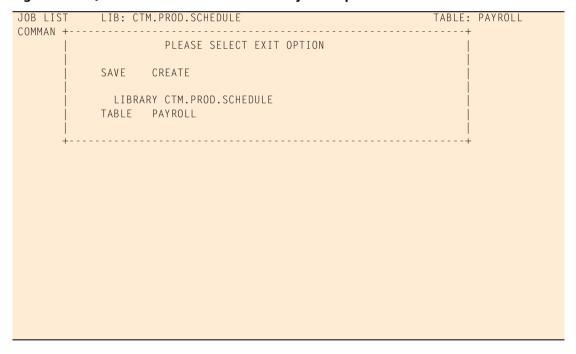
Option	Description
В	Indicates that the target of a copy or move is directly before this line.
С	Copy this line to the target.
M	Move this line to the target.

After performing requested options, CONTROL-M automatically handles renumbering and adjusts the relevant DEPENDS ON parameter values on the screen.

# Step 4: Exit the Quick Schedule Definition Facility (and Create the Scheduling Table)

To exit the Quick Schedule Definition facility after entering the data for a table, press the END (PF03/PF15) key. An Exit Option window is opened:

Figure 131 Quick Schedule Definition Facility Exit Option Window



The schedule can be saved (to replace a table of the same name that previously existed in the library), or created (to store a new table in the library), by typing Y in the appropriate exit option. The job schedule is automatically created as you exit.

If N is entered, the table is not saved, and the schedule is not produced. You return to the Utilities screen or other screen, depending on how you entered the utility.

If no changes have been made, the Exit Option window is not opened.

To exit to the Quick Schedule Definition entry panel without saving your entries (and without creating the job schedule), press RESET (**PF04**).

The screen below illustrates job GOVTREPT selected from the jobs listed in the Job List screen in Step 3 above. Note particularly the automatically created MEMNAME, IN, and OUT parameters, and the job name inserted into the SHOUT message in place of the %%JOBNAM variable in the skeleton job definition.

Figure 132 Scheduling Definition Screen Quick Schedule Definition Example

```
JOB: GOVTREPT LIB CTM.PROD.SCHEDULE
                                                                 TABLE: BACKUP
                                                                SCROLL===> CRSR
COMMAND ===>
  MEMNAME GOVTREPT MEMLIB CTM.PROD.JOBLIB
  OWNER M44 TASKTYPE JOB PREVENT-NCT2 DFLT N
APPL APPL-L GROUP BKP-PROD-L
  DESC REPORTS TO GOVERNMENT
  OVERLIB
                                                            STAT CAL
                               SYSTEM ID
                                                            NJF NODE
  SCHENV
  SET VAR
  CTB STEP AT NAME
DOCMEM GOVTREPT DOCLIB
                                 TYPE
                                                                      AND/OR O
                                                                 WCAL
  WDAYS
  MONTHS 1- N 2- N 3- N 4- N 5- N 6- N 7- Y 8- N 9- Y 10- N 11- N 12- N
  DATES
                   SHIFT RETRO N MAXWAIT 00 D-CAT
  CONFCAL
  MINIMUM
                   PDS
  DEFINITION ACTIVE FROM UNTIL
  IN FINANCE-CHECKCAL-OK ODAT
  CONTROL
  RESOURCE
  FROM TIME + DAYS UNTIL TIME + DAYS
DUE OUT TIME + DAYS PRIORITY 00 SAC CONFIRM
  TIME ZONE:
  OUT FINANCE-GOVTREPT-OK ODAT +
AUTO-ARCHIVE Y SYSDB Y MAXDAYS MAXRUNS
RETENTION: # OF DAYS TO KEEP # OF GENERATIONS TO KEEP
  SYSOUT OP (C,D,F,N,R)

MAXRERUN RERUNMEM

STEP RANGE FR (PGM.PROC)
                                                                        FROM
                                                   INTERVAL
TO
                                                                    FROM
  ON PGMST ANYSTEP PROCST CODES NOTOK

DO SHOUT TO TSO-M44 URGENCY R

= JOB GOVTREPT ENDED "NOT OK"
                                                                         A/0
  SHOUT WHEN TIME + DAYS TO
                                                                      URGN
COMMANDS: EDIT, DOC, PLAN, JOBSTAT
                                                                        20.28.53
```

## **M6: End-User Job Order Interface**

By ordering jobs through the End-User Job Order Interface, you can bypass the rest of the CONTROL-M Online facility.

The Job List screen (Figure 133) can be displayed in one of the following ways:

- Select option M6 on the Online Utilities menu.
- Activate CLIST CTMJBINT from the TSO Command Processor.

Figure 133 Job List Screen Entered Through the End-User Job Order Interface

This screen displays a list of jobs that the particular user is permitted to order. The INCONTROL administrator determines which jobs each user is permitted to order.

- 1. Do one of the following:
- Press **PF03/PF15** to exit the screen.
- To order jobs, type S in the OPT field to the left of each job to be ordered and press Enter.

For each job selected, in sequence, a new screen is displayed. The header of the screen contains the names of the library and table. Under the command line, the Please Enter Scheduling Date/Force Options window is displayed (see Figure 134).

Figure 134 Job Scheduling Date and FORCE Options Window

- 2. In this window, you can
- replace the date displayed with another date
- change the scheduling date of the job to the CONTROL-M working date, by typing Y in the OR USE CONTROL-M WORKING DATE field
- FORCE the job by typing YES in the FORCE JOB? field
- wait for a specific date by typing YES in the WAIT FOR ODATE? field, or process the Order/Force request immediately by typing NO.
- 3. After selecting these options, you have the following further options:
- Press **Enter** to complete the order request.
- Press **PF03/PF15** to cancel the order request.
- Press **PF04/PF16** to cancel the changes and exit the Job List screen.

## **U1: Invoke DOCU/TEXT**

This option is available only at sites that have installed DOCU/TEXT, a product of Diversified Systems Software, Inc. DOCU/TEXT provides automated, online JCL documentation, and this option provides a direct interface to DOCU/TEXT. It can be activated by requesting option U1 on the Online Utilities menu or by activating CLIST CTMCDOCU directly.

For details about product usage refer to your DOCU/TEXT user guide.





# **Job Production Parameters**

This chapter includes the following topics:
Overview
General Parameters – Summary
Basic Scheduling Parameters – Summary
Runtime Scheduling Parameters – Summary
Post-Processing Parameters – Summary
Parameter Descriptions
ADJUST CONDITIONS: General Job Parameter
APPL: General Job Parameter
APPL FORM: General Job Parameter
APPL TYPE: General Job Parameter
APPL VER: General Job Parameter
§Restart§ AUTO-ARCHIVE: Post–Processing Parameter
CM VER: General Job Parameter
CONFCAL: Basic Scheduling Parameter
CONFIRM: Runtime Scheduling Parameter
CONTROL: Runtime Scheduling Parameter
CTB STEP: General Job Parameter
D-CAT: Basic Scheduling Parameter
DATES: Basic Scheduling Parameter
DAYS: Basic Scheduling Parameter
DEFINITION ACTIVE: Basic Scheduling Parameter
DESC: General Job Parameter
DO Statement: Post–Processing Parameter
DO COND: Post-Processing Parameter
DO CTBRULE: Post–Processing Parameter
DO FORCEJOB: Post–Processing Parameter
§Restart§DO IFRERUN: Post–Processing Parameter
DO MAIL: Post-Processing Parameter
DO NOTOK: Post-Processing Parameter
DO OK: Post–Processing Parameter
DO REMEDY: Post–Processing Parameter
DO RERUN: Post-Processing Parameter
DO SET: Post–Processing Parameter
DO SHOUT: Post–Processing Parameter

DO STOPCYCL: Post-Processing Parameter	. 475
DO SYSOUT: Post-Processing Parameter	. 477
DOC: General Job Parameter	. 486
DOCLIB: General Job Parameter	. 488
DOCMEM: General Job Parameter	. 490
DUE OUT: Runtime Scheduling Parameter	. 492
GROUP: General Job Parameter	. 495
GRP MAXWAIT: Basic Scheduling Parameter	. 498
IN: Runtime Scheduling Parameter	
INSTREAM JCL: General Job Parameter	. 511
INTERVAL: Post-Processing Parameter	. 513
MAXRERUN: Post-Processing Parameter	. 516
MAXWAIT: Basic Scheduling Parameter	
MEMLIB: General Job Parameter	
MEMNAME: General Job Parameter	. 528
MINIMUM: Basic Scheduling Parameter	. 531
MONTHS: Basic Scheduling Parameter	. 533
NJE NODE: General Job Parameter	. 535
ON Statements: Post-Processing Parameter	. 537
ON GROUP-END: Post-Processing Parameter	. 539
ON PGMST: Post-Processing Parameter	
ON SYSOUT: Post-Processing Parameter	
OUT: Post-Processing Parameter	. 562
OVERLIB: General Job Parameter	
OWNER: General Job Parameter	
PDS: Basic Scheduling Parameter	
PIPE: General Job Parameter	
\$Restart\$PREVENT-NCT2:General Job Parameter	
PRIORITY: Runtime Scheduling Parameter	
RELATIONSHIP: Basic Scheduling Parameter	
RERUNMEM: Post-Processing Parameter	
RESOURCE: Runtime Scheduling Parameter	. 596
RETENTION: # OF DAYS TO KEEP: Post–Processing Parameter	. 603
RETENTION: # OF GENERATIONS TO KEEP: Post-Processing Parameter	. 605
RETRO: Basic Scheduling Parameter	
SAC: Run Time Parameter	
SCHEDULE TAG: Basic Scheduling Parameter	
SCHEDULE TAG ACTIVE: Basic Scheduling Parameter	. 616
SCHENV: General Job Parameter	
SET VAR: General Job Parameter.	
SHOUT: Post-Processing Parameter	
STAT CAL: General Job Parameter	
STEP RANGE: Post-Processing Parameter	. 641
SYSOUT: Post-Processing Parameter	
SYSTEM ID: General Job Parameter	
TASKTYPE: General Job Parameter	
TIME + DAYS: Runtime Scheduling Parameter	
TIME ZONE: Runtime Scheduling Parameter	
WDAYS: Basic Scheduling Parameter	669

# **Overview**

Job scheduling definitions consist of parameters that correspond to the decisions made and actions performed when handling the scheduling, submission and post-processing of a job. Job scheduling definitions are defined in the Job Scheduling Definition screen, which is shown in Figure 135. This is the main screen of the Scheduling Definition facility.

Figure 135 Job Scheduling Definition Screen

```
JOB: BACKPLO2 LIB CTM.PROD.SCHEDULE
                                                                TABLE: BACKUP
COMMAND ===>
                                                                SCROLL==> CRSR
  MEMNAME BACKPLO2 MEMLIB CTM.PROD.JOBLIB

OWNER M44 TASKTYPE JOB PREVENT-NCT2 Y DFLT N

APPL APPL APPL-1 GROUP BKP-PROD-1
  APPL
         APPL-L
                                     GROUP BKP-PROD-L
         DAILY BACKUP OF SPECIAL FILES FROM APPL-L
  OVERLIB CTM.OVER.JOBLIB
                                                            STAT CAL
                                SYSTEM ID
                                                            NJE NODE
  SCHENV
  SET VAR
  CTB STEP AT NAME
                                     TYPF
  DOCMEM BACKPLO2 DOCLIB CTM.PROD.DOC
  SCHEDULE TAG
  RELATIONSHIP (AND/OR) 0
                                                                DCAL
  DAYS
                                                                     AND/OR
  WDAYS
                                                                WCAL
  MONTHS 1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- Y 11- Y 12- Y
  DATES
  CONFCAL WORKDAYS SHIFT
                               RETRO N MAXWAIT 04 D-CAT
  MINIMUM
  DEFINITION ACTIVE FROM
                                 UNTII
           START-DAILY-BACKUP ODAT
  CONTROL
  RESOURCE INIT
                               0001 CART
                                                               0001
  PIPE CTM.PROD.PIPE
  FROM TIME + DAYS
DUE OUT TIME + DAYS
                               UNTIL TIME
                                 PRIORITY
  TIME ZONE:
        BAKCKPLO2-ENDED-OK ODAT +
  OUT
  AUTO-ARCHIVE Y SYSDB Y MAXDAYS
                                                      MAXRUNS
  RETENTION: # OF DAYS TO KEEP 030 # OF GENERATIONS TO KEEP
  SYSOUT OP (C,D,F,N,R)
                                                                       FROM
              RERUNMEM
                                                     INTERVAL
  MAXRERUN
                                                                   FROM
  STEP RANGE
                   FR (PGM.PROC)
                                   CODES
  ON PGMST
                   PROCST
                                                                        A/0
    D0
                                                     FROM 001 TO 132
  ON SYSOUT
                                                                        A/0
    DO
  SHOUT WHEN
                     TIME
                                                T0
                                                                     URGN
                                       DAYS
```

```
APPL TYPE

APPL FORM

INSTREAM JCL: N

-----

COMMANDS: EDIT, DOC, PLAN, JOBSTAT

APPL VER

CM VER

TOMMANDS: EDIT, DOC, PLAN, JOBSTAT

APPL VER

CM VER

TOMMANDERS ***

19.17.13
```

#### NOTE -



The SCHEDULE TAG and RELATIONSHIP fields only appear in job scheduling definitions belonging to Group scheduling tables.

The PIPE parameter is displayed only if MAINVIEW Batch Optimizer (MVBO) is installed.

RETENTION parameters # OF DAYS TO KEEP and # OF GENERATIONS TO KEEP are displayed only at sites that use the History Jobs file.

If the scheduling table is a Group scheduling table, a Group Entity (shown in Figure 136) must be defined before the job scheduling definitions.

Figure 136 Group Entity Definition Screen

```
GRP ACCOUNTS_GROUP CTM.PROD.SCHEDULE(GRP)
COMMAND ===>
                                                   SCROLL===> CRSR
 GROUP ACCOUNTS_GROUP MEMNAME ACCOUNTS
 OWNER NO4B
 APPL
 DESC
 ADJUST CONDITIONS N
                          GRP MAXWAIT 00
                                           STAT CAL
  SET VAR
 DOCMEM ACCOUNTS DOCLIB CTM.PROD.DOC
         _____
 SCHEDULE TAG ALL_DAYS
                                                   DCAL
 DAYS
      ALL
                                                       AND/OR
 WDAYS
                                                   WCAL
 MONTHS 1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- Y 11- Y 12- Y
 SCHEDULE TAG ACTIVE FROM INTIL
 DATES
 SCHEDULE TAG
 DAYS
                                                   DCAL
                                                       AND/OR
 WDAYS
                                                   WCAL
 MONTHS 1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- Y 11- Y 12- Y
 DATES
          SHIFT
                         RETRO N MAXWAIT 00
 SCHEDULE TAG ACTIVE FROM
                           UNTII
 ΙN
 CONTROL
              + DAYS UNTIL TIME + DAYS
 FROM TIME
 DUE OUT TIME + DAYS
                           PRIORITY SAC CONFIRM
  TIME ZONE:
 OUT
```

Ī

Most parameters in the Group Entity definition are the same as in the job scheduling definition, but they apply to the group as a whole. Therefore:

- At least one set of basic scheduling criteria in the Group Entity must be satisfied before any job in the group can be scheduled.
- Runtime scheduling criteria in the Group Entity must be satisfied before any job in the group can be executed.
- Post-processing statements in the Group Entity are applied only after all jobs in the group have finished executing.

The following parameters in the Group Entity are not found in the job scheduling definition:

- ADJUST CONDITIONS
- ON GROUP-END
- SCHEDULE TAG ACTIVE FROM and UNTIL

Usage and operation of the Scheduling Definition facility, including entry and exit of the Job Scheduling Definition screen, is described in the *INCONTROL* for *z/OS Utilities Guide*.

In addition to defining jobs through the Scheduling Definition facility, jobs can also be defined using the batch utility CTMBLT, described in the *INCONTROL for z/OS Utilities Guide*, or using the online utility QUICKDEF. The QUICKDEF utility, the Quick Schedule Definition facility, is available under ISPF only. For more information about it, see "Utilities Under ISPF" on page 323.

This chapter provides a detailed description of the job scheduling definition parameters and statements.

The parameters of the Job Scheduling Definition screen are divided into the following categories:

- General parameters
- Basic Scheduling parameters
- Runtime Scheduling parameters
- Post-processing parameters

A brief summary of the parameters in each category is provided on the following pages. This is followed by a detailed description of each parameter, in alphabetical order.

# **General Parameters – Summary**

General parameters provide general information about the job and certain information required by the JCL. Information about the following parameters is provided in this chapter:

Table 152 General Parameters (part 1 of 2)

Parameter	Description
MEMNAME	Member containing the JCL.
MEMLIB	Library containing the JCL member.
OWNER	Owner of the job.
TASKTYPE	Type of job or task.
<b>§Restart§</b> PREVENT NCT2	Whether to perform data set cleanup before the original job run.
	DFLT – Protected field indicating the PREVENT-NCT2 default value of the site.
APPL	Application to which the job belongs.
GROUP	Group to which the job belongs.
DESC	Brief description of the job.
OVERLIB	Library containing a special case JCL for the job.
STAT CAL	Name of a periodic calendar that will be used to gather average runtime statistics for the job, based on a period.
SCHENV	Name of the workload management scheduling environment to be associated with the job.
SYSTEM ID	In JES2, the identity of the system in which the job must be initiated and executed.
	In JES3, the identity of the processor on which the job must be executed.
NJE NODE	Identifies the node in the JES system at which the job must execute.
SET VAR	Mechanism for setting the value of a JCL user-defined variable.
CTB STEP	CONTROL-M/Analyzer step to be added to the execution of the job.
DOCMEM	Member containing detailed information about the job.
DOCLIB	Library containing the member specified in the DOCMEM parameter.
DOC	Detailed job documentation.

Table 152 General Parameters (part 2 of 2)

Parameter	Description
INSTREAM JCL	Whether CONTROL-M for z/OS submits a JCL stream defined within the job scheduling definition.
The following General parameters are in the Group Entity only:	
ADJUST CONDITIONS	Allows conditions to be removed from job orders if the predecessor jobs that set the conditions are not scheduled.
GRP MAXWAIT	Number of additional days after the original scheduling date that the Group Entity can remain in the Active Jobs file if it does not have a status of ENDED OK (and none of its jobs currently appear in the Active Jobs file).

# **Basic Scheduling Parameters – Summary**

Basic Scheduling parameters determine if the job is a candidate for execution on a specific date. If a job is a candidate for execution on a specific date, a job order is automatically placed in the Active Jobs file during New Day processing.

Each job order placed in the Active Jobs file is associated with an original scheduling date. This is the date the job is to run according to the Basic Scheduling parameters. This date is not necessarily the same as the current system date or the current working date. For further information, see "Date Definition Concepts" on page 63

Basic Scheduling parameters and subparameters allow different methods of expressing a job schedule.

The SCHEDULE TAG parameter appears only in Group tables, in both job scheduling definitions and in the Group Entity.

Each set of basic scheduling criteria in the Group Entity must be uniquely labeled by a SCHEDULE TAG value. At least one Schedule Tag must be defined.

In job scheduling definitions, SCHEDULE TAG is optional. Each specified SCHEDULE TAG value in the job scheduling definition must match a SCHEDULE TAG value in the Group Entity. The associated basic scheduling criteria can then be applied to the job.

The RELATIONSHIP parameter appears only in job scheduling definitions in Group scheduling tables.

The RELATIONSHIP parameter defines the relationship (AND/OR) between schedule tag criteria and the basic scheduling criteria of the job, that is, whether one or both sets of criteria are to be satisfied.

The Basic Scheduling parameters, except SCHEDULE TAG and RELATIONSHIP, are listed below by category. When defining basic scheduling criteria for jobs in regular or Group scheduling tables, or when defining basic scheduling criteria for Group Entities, the following rules apply to these categories of parameters:

- Parameters must be selected from one and only one of the first three categories (A, B, or C).
- Parameters in the last two categories (D and E) are optional.

Table 153 Category A, B, C, and D Parameters

Category	Parameter
A	■ MONTHS – Schedule the job during the specified months.
	<ul> <li>DAYS – Schedule the job on specified days (in the above-specified months) and/or select days from a specified calendar.</li> </ul>
	■ WDAYS – Schedule the job on specified days of the week (in the above-specified months) and/or select days from a specified calendar.
	<ul> <li>CONFCAL – Confirm scheduling days against a specified calendar.</li> </ul>
В	■ DATES – Schedule the job on specified dates.
	■ WDAYS – Schedule the job on specified days of the week.
	<ul> <li>CONFCAL – Confirm scheduling days against a specified calendar.</li> </ul>
С	■ PDS – PDS to be checked for minimum number of tracks.
	■ MINIMUM – Minimum number of tracks.
D	RETRO – Schedule the job even if the original scheduling date has passed.
E	MAXWAIT – Maximum number of days to keep the job in the Active Jobs file awaiting execution after its original scheduling date has passed.
	■ D-CAT – CONTROL-D category of the job. (Documented as CATEGORY prior to version 5.1.4.)

### **Category C Parameters**

Schedule the job if the number of free tracks in the specified partitioned data set (PDS) is less than the minimum number of tracks specified. This set of criteria is intended for jobs or started tasks that clean, compress or enlarge libraries or that issue warning messages if the minimum number of free tracks is not available.

## **Basic Scheduling Parameters**

Each Basic Scheduling parameter is described in this chapter. However, the interrelationships between some of these parameters are described briefly below.

### DAYS, DCAL, WDAYS, WCAL

These parameters are all optional.

The DAYS parameter identifies days of the month on which the job must be scheduled, such as first day of the month, third working day of the month. Several formats are available for specifying DAYS values.

The WDAYS parameter identifies days of the week on which the job must be scheduled, such as the first day of the week, the second day of each week, and so on. Several formats are available for specifying WDAYS values.

A calendar name can be specified in the DCAL and/or WCAL fields. A calendar specifies days of the year on which a rule can be scheduled, known as working days. For more information on calendars and the IOA Calendar facility, see "IOA Calendar Facility" on page 306.

When both the DAYS and DCAL parameters are specified, they work as a complementary unit, as described in "DAYS: Basic Scheduling Parameter" on page 420. Similarly, when both WDAYS and WCAL are specified, they also work as a complementary unit, as described in "WDAYS: Basic Scheduling Parameter" on page 669.

When values for both DAYS (/DCAL) and WDAYS (/WCAL) are specified in the same job scheduling definition, the resulting schedule is determined by the value specified in field AND/OR.

#### CONFCAL and SHIFT

A calendar specified in CONFCAL is not used for job scheduling, but is used instead for validating a scheduled date. Only jobs that have satisfied all other specified basic scheduling criteria are checked against the CONFCAL calendar. If the day is a working day in the CONFCAL calendar, the job is scheduled on that day. Otherwise, the job is either shifted to (scheduled on) another day according to the value entered in the SHIFT parameter, or the job is not scheduled (if no SHIFT value has been specified).

CONFCAL calendars are especially useful for handling holidays and other scheduling exceptions.

### **Defining a Schedule – Internal Scheduling Logic**

When defining scheduling tables, it is useful to understand the IOA Scheduling facility logic, which determines whether to order a job on a specific day. This logic is described below.

- 1. ACTIVE FROM and UNTIL parameters are checked first. If the current date falls outside the time range specified, no further checking is performed.
- 2. DAYS and DCAL parameters are checked independently and a first tentative scheduling decision is created.
- 3. WDAYS and WCAL parameters are checked independently and a second tentative scheduling decision is created.
- 4. A third tentative scheduling decision is created based on the above two decisions and the AND/OR value linking them.
  - (If DAYS and/or DCAL are not specified, this third temporary scheduling decision is identical to the second scheduling decision. If WDAYS and/or WCAL are not specified, this third scheduling decision is identical to the first scheduling decision.
- 5. If CONFCAL and/or SHIFT are specified, this third scheduling decision is adjusted according to the CONFCAL and SHIFT criteria.
- This third scheduling decision (as adjusted) becomes the final scheduling decision.The IOA Scheduling facility determines whether to schedule a job based on this final scheduling decision.

### Scheduling Jobs in Group Scheduling Tables

The following scheduling algorithm applies to Group scheduling tables:

- 1. Before jobs in a group can be scheduled, the group must be eligible for scheduling (that is, at least one of the tagged sets of basic scheduling criteria in the Group Entity has been satisfied).
- 2. If (and only if) the group is eligible for scheduling, each job scheduling definition in the scheduling table is individually checked for possible scheduling.
- 3. For each job scheduling definition:
  - Schedule tags in the job scheduling definition are checked sequentially beginning with the first tag. The SCHEDULE TAG ACTIVE FROM and UNTIL parameters are checked first. If the current date falls outside the time range specified, no further checking is performed.

If the criteria of a schedule tag are satisfied, no further checks are performed on the remaining schedule tags. The criteria belonging to the satisfied tag are used in the scheduling algorithm.

- The RELATIONSHIP parameter (AND/OR) is checked.
  - If a schedule tag was satisfied and the defined relationship is OR, the satisfied schedule tag is sufficient and the job is scheduled according to criteria of this tag. No further checks are performed.
  - If no schedule tag was satisfied and the defined relationship is AND (that is, the job requires that the schedule tag be satisfied), the job is not scheduled. No further checks are performed.
  - If a schedule tag was satisfied and the defined relationship is AND, or if no schedule tag was satisfied and the defined relationship is OR, the basic scheduling criteria of the job must be satisfied (that is, the algorithm continues with the next step).
- The basic scheduling criteria of the job are checked.
  - If the basic scheduling criteria of the job are not satisfied, the job is not scheduled.
  - If the basic scheduling criteria of the job are satisfied, the job is scheduled.

The basic scheduling criteria of the job, not the scheduling tag criteria, are used for scheduling. This is a concern only if there are conflicting MAXWAIT values in the scheduling tag criteria and the basic scheduling criteria of the job. In this case, the MAXWAIT value from the basic scheduling criteria of the job is used.

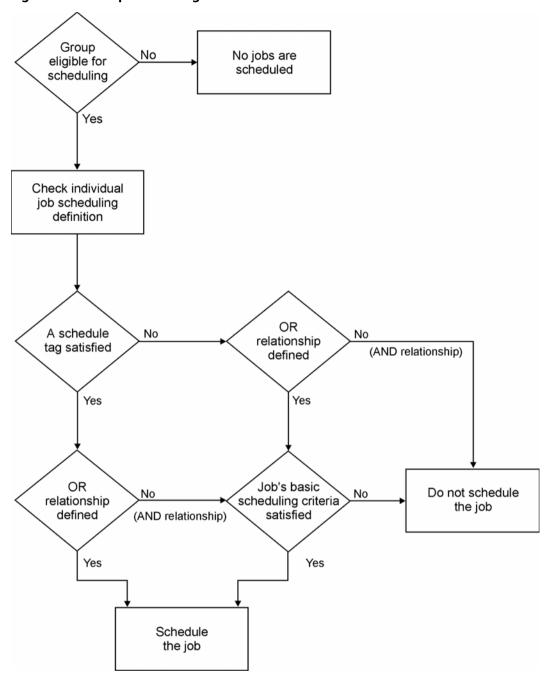


Figure 137 Group Scheduling Flowchart

# **Runtime Scheduling Parameters – Summary**

Runtime Scheduling parameters define job submission criteria. The job is not submitted unless all submission criteria are satisfied. The following criteria can be defined:

**Table 154 Runtime Scheduling Parameter Criteria** 

Parameter	Description
IN	Required prerequisite conditions.
CONTROL	Required exclusive or shared Control resources.
RESOURCE	Quantitative resources and the required quantity.
PIPE	Name of each data set that is replaced by a pipe during the run of the job. Available only at sites where MAINVIEW Batch Optimizer (MVBO) is installed.
TIME + DAYS	Time range during which the job must be submitted.
TIME ZONE	Enables automatic adjustment of the times specified in the job definition to the corresponding times in a different time zone.
PRIORITY	Job priority and critical path priority.
DUE OUT	Time by which the job must finish executing, which can determine the time by which the job must be submitted.
CONFIRM	Manual confirmation required before the job is submitted.

# **Post-Processing Parameters – Summary**

Actions to be performed after job execution generally depend on the results of the job execution:

- Certain actions may be required when the job ends successfully.
- Certain actions may be required when the job fails, depending on the reason for failure.
- Certain actions may be required in any and all situations.

The CONTROL-M monitor tracks each job execution. Following the termination of a job, the CONTROL-M monitor checks the execution results of each step in the job. Based on the results, the CONTROL-M monitor determines a final status of the job. Either of two final job statuses can be assigned:

Table 155 Final Job Statuses

Status	Description
OK	Job ended OK. This status is usually assigned when all steps in the job end with a condition code less than or equal to C0004.
NOTOK	Job ended NOTOK. This status is assigned when any step ends with a condition code greater than or equal to C0005. It is also assigned if the job abends or is not run.
	The following statuses are subsets of end status NOTOK:
	■ JNRUN – Job not run due to JCL syntax error.
	■ EXERR – Execution error (that is, one that occurred after the job has started running).
	■ JFAIL – JCL error was encountered during job step initiation. This status is also a subset of status EXERR.

If a post-processing error occurs after a job ends OK (including FORCE OK), it indicates that there is a problem with the post-processing statements defined in the job scheduling definition. For example, a post-processing statement may have indicated an action that the owner of the job was not authorized to perform.

Post-processing parameters can be divided into the following groups:

#### Parameters Performed When the Job Ends OK

Table 156 Parameters Performed When the Job Ends OK

Parameter	Description
OUT	Adds or deletes prerequisite conditions.
<b>§Restart§</b> AUTO-ARCHIVE	Archives sysout.
RETENTION	Specifies retention criteria of a job in the History Jobs file.
SYSOUT	Specifies sysout processing.

### **Conditional Processing or Processing in All Situations**

Most conditional processing is specified through a combination of ON and DO statements. ON and DO statement definition consists of defining ON statement step and code events (for example, ON PGMST STEP1 CODE C0016), followed by DO statement actions (for example, DO SHOUT, DO FORCEJOB), which are performed when the ON step and code criteria are satisfied.

A range of steps for use in the ON statement can be defined in the STEP RANGE parameter.

ON and DO statements also specify actions that are to be performed in any and all cases. To ensure that the ON statement is activated for all step and code events, enter reserved word ANYSTEP as the ON step name and ***** as the ON code.

**Table 157 Conditional Processing Statements** 

DO Statement	Description
DO statements allow s criteria are satisfied:	pecification of a wide variety of actions to be performed when the ON
DO COND	Add or delete prerequisite conditions.
DO CTBRULE	Activate a CONTROL-M/Analyzer rule.
DO FORCEJOB	Force a job.
§Restart§ DO IFRERUN	Perform CONTROL-M/Restart job restart.
DO MAIL	Send an e-mail to the specified recipients.
DO NOTOK	Set the status of the step to NOTOK.
DO OK	Set the status of the step to OK.
DO REMEDY	Open a Remedy Helpdesk ticket.
DO RERUN	Rerun the job.
DO SET	Set the value of an AutoEdit variable.
DO SHOUT	Send a message.
DO STOPCYCL	Stop recycling a cyclic task.
DO SYSOUT	Handle sysout processing.
The following parameter performed.	ter specifies the condition that determines if and when processing is
SHOUT	Sends a message to a specified destination in specified situations (for example, if the job was submitted late).

### **Return and Cyclic Post-processing Parameters**

**Table 158 Return and Cyclic Post-Processing Parameters** 

Parameter	Description
MAXRERUN	Maximum number of times to rerun the job (used only for automatic job rerun or cyclic jobs). (Called RERUN – MAXRERUN prior to version 6.0.00).
RERUNMEM	Member containing the JCL to be used for automatic rerun. (Called RERUN – RERUNMEM prior to version 6.0.00.)
INTERVAL	Minimum time interval between runs of a rerun or cyclic job. This parameter acts as a Runtime Scheduling parameter for the subsequent rerun or cyclic runs of the job.

## **Group Entity Post-processing Parameters**

**Table 159 Group Entity Post-Processing Parameters** 

Parameter	Description
The following parame	eter is found only in the Group Entity definition:
ON GROUP-END	The table-processing termination status that determines whether the accompanying DO statements are performed.
	The following DO statements are permitted following an ON GROUP-END statement:
	<ul> <li>■ DO COND</li> <li>■ DO OK</li> <li>■ DO MAIL</li> <li>■ DO FORCEJOB</li> <li>■ DO SET</li> <li>■ DO NOTOK</li> <li>■ DO SHOUT</li> </ul>

#### NOTE



DO OK and DO NOTOK statements change the final status of the group (not the status of each job or job step in the group).

# **Parameter Descriptions**

The following pages contain detailed descriptions of all parameters available in the CONTROL-M Job Definition screen. Parameters are arranged in alphabetical order. Within each parameter, subparameters are arranged according to the order of the fields on the screen.

Each parameter begins on a new page, including:

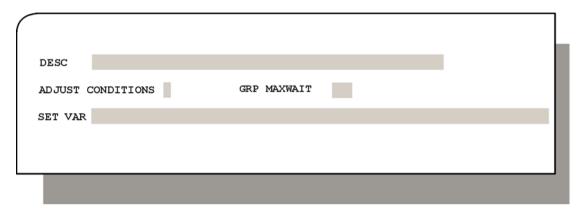
- A brief explanation of the purpose of the parameter.
- The format required for defining the parameter within an extract of the CONTROL-M screen.
- General information explaining the parameter and its usage.
- Where applicable, some practical examples illustrating implementation of the parameter.

For more information on the Job Definition facility, see Chapter 2, "Online Facilities."

# **ADJUST CONDITIONS: General Job Parameter**

Determines how CONTROL-M handles a requirement for a prerequisite condition by successor jobs in a Group scheduling table. This parameter appears in the Group Entity only and applies only to Group scheduling tables.

Figure 138 ADJUST CONDITIONS Parameter Format



Optional. Valid values are shown in Table 160.

Table 160 ADJUST CONDITIONS Parameter Values

Value	Description
Y (Yes)	CONTROL-M ignores, for each job in the group, any IN prerequisite condition that is triggered by a predecessor job that will not be ordered during the current day.
N (No)	CONTROL-M does not erase any IN prerequisite conditions. Default.
D (Dummy)	CONTROL-M orders as a DUMMY job any job with scheduling criteria that are not satisfied on the current ODATE.

### **General Information**

This parameter is applied to all jobs in the Group scheduling table. It provides the options shown in Table 160 when job dependency is defined between jobs in a group by their respective conditions.

Such a job dependency arises when one job in a group is dependent on a condition that is triggered by another job in the group. If the triggering job is not ordered, meaning, placed on the Active Jobs file, because the scheduling criteria of the triggering job do not match the current Odate, the dependent job cannot run.

The options provided by the ADJUST CONDITIONS parameter enable you to avoid this difficulty.

- If you enter the value Y for the ADJUST CONDITIONS parameter, the following principles apply:
  - CONTROL-M erases IN conditions from each job in the group that is placed on the AJF. The following criteria must be met before an IN condition is erased:
    - The IN condition in the job is triggered by a predecessor job.
    - The scheduling criteria of the predecessor job are such that the predecessor job will not be ordered during the current day.

The erasing of these IN conditions frees the successor job from its dependency on the predecessor job.

- Only the image of the job on the AJF is affected. The original job scheduling definition remains unchanged.
- The erased condition does not appear in the Zoom screen.
- If you enter the value N for the ADJUST CONDITIONS parameter, CONTROL-M runs normally. You must release any jobs that are likely to wait indefinitely on the AJF because they are dependent on predecessor jobs with scheduling criteria such that they will not be ordered. To release the dependent jobs, use one of the manual options, such as those available in
  - the IOA Conditions/Resources screen (Screen 4)
  - the IOA Manual Conditions file (Screen 7)
- If you enter the value D for the ADJUST CONDITIONS parameter, the following principles apply:
  - CONTROL-M places all the jobs in the group onto the AJF when the group is ordered.
  - Jobs in the group with scheduling criteria that fit the current ODATE remain unchanged.
  - Jobs in the group with scheduling criteria that do not fit the current ODATE, and would not ordinarily be ordered, are placed on the AJF as DUMMY jobs (the MEMLIB parameter is changed to DUMMY). This permits the original logical flow of the group to be maintained by preserving the relationships of the IN and OUT conditions of these dummy jobs.
  - Only the image of the job on the AJF is affected. The original job scheduling definition remains unchanged.

## **Examples**

### **Example 1**

If a predecessor job is not scheduled, the requirement for the prerequisite conditions that the predecessor job would have normally placed in the IOA Conditions file is erased in the successor jobs.

Figure 139 ADJUST CONDITIONS Parameter Example

```
GRP ACCOUNTS_GROUP CTM.PROD.SCHEDULE(GRP)
COMMAND ===>
                                                             SCROLL===> CRSR
  GROUP ACCOUNTS_GROUP MEMNAME ACCOUNTS
  OWNER NO4B
  APPI
  DESC
  ADJUST CONDITIONS Y
                               GRP MAXWAIT 00
  DOCMEM ACCOUNTS DOCLIB CTM.PROD.DOC
  SCHEDULE TAG ALL_DAYS
  DAYS ALL
                                                             DCAL
                                                                 AND/OR
  WDAYS
                                                             WCAL
  MONTHS 1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- Y 11- Y 12- Y
  DATES
  CONFCAL SHIFT RETRO N MAXWAIT 00 SCHEDULE TAG ACTIVE FROM UNTIL
  SCHEDULE TAG SUNDAYS
                                                             DCAL
  DAYS
        0.1
                                                                  AND/OR
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
                                                                   16.44.31
```

### **Example 2**

Assume that a Group Entity defines a chain of jobs, in the order DAILY-1 —> DAILY-2 —> DAILY-3 —> WEEKLY-1 —> DAILY-4.

- The user wants the following:
  - Each of the DAILY jobs must run every day, in the order DAILY-1 —> DAILY-2 —> DAILY-3 —> DAILY-4.
  - The WEEKLY-1 job must run only on each Tuesday.
  - When WEEKLY-1 runs, it must run after DAILY-3 and before DAILY-4.
- DAILY-1, DAILY-2, DAILY-3, WEEKLY-1, and DAILY-4 contain IN and OUT conditions to ensure that they run in the required order.

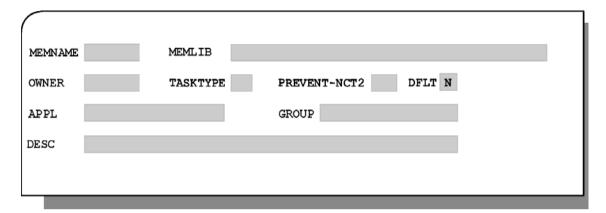
Unless some step is taken, on the days when WEEKLY-1 does not run, DAILY-4 cannot be executed, because its IN condition causes it to wait for WEEKLY-1 to end.

If the ADJUST CONDITIONS parameter is set to D, WEEKLY-1 is ordered daily. Every day except Tuesday, its scheduling conditions prevent it running, and it is ordered as a PSEUDO job. Each Tuesday, it is ordered and executed normally. The logical job flow is maintained throughout.

# **APPL: General Job Parameter**

Descriptive name of the application to which the group of the job belongs. Used as a common descriptive name for a set of related groups (of jobs).

Figure 140 APPL Parameter Format



APPL specifies an application name of 1 through 20 characters. Only trailing blanks are allowed.

By default, the APPL parameter is optional. It can be modified in the user profile.

## **General Information**

The parameter facilitates the handling of groups of production jobs.



### NOTE

Use of the APPL parameter is highly recommended to facilitate implementation of CONTROL-M/Enterprise Manager functions and future CONTROL-M options. For details, see the *CONTROL-M/Enterprise Manager User Guide*.

## **Example**

Job OPERCOMP belongs to group MAINTENANCE, which is part of application OPER.

### Figure 141 APPL Parameter Example

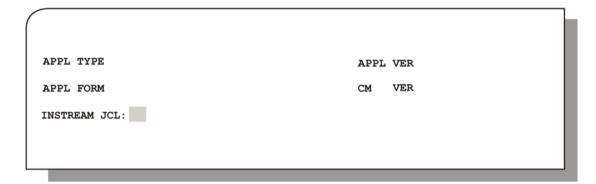
```
JOB: OPERCOMP LIB CTM.PROD.SCHEDULE
                                                            TABLE: OPER
                                                            SCROLL===> CRSR
COMMAND ===>
  MEMNAME OPERCOMP MEMLIB GENERAL
  OWNER M44 TASKTYPE JOB PREVENT-NCT2 Y DFLT N APPL OPER GROUP MAINTENANCE
  DESC JOB RUN ON THE 1ST OF THE MONTH
  OVERLIB
                                                         STAT CAL
  SCHENV
                             SYSTEM ID
                                                        NJE NODE
  SET VAR
  CTB STEP AT NAME TYPE
DOCMEM OPERCOMP DOCLIB CTM.PROD.DOC
  DAYS 01
                                                             AND/OR
  WDAYS
                                                             WCAL
  MONTHS 1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- Y 11- Y 12- Y
  DATES
                 SHIFT
  CONFCAL
                           RETRO Y MAXWAIT OO D-CAT
  MINIMUM
                 PDS
  DEFINITION ACTIVE FROM UNTIL
  ΙN
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
                                                         16.44.00
```

# **APPL FORM: General Job Parameter**

The type of form used for entering application data.

This is a protected field whose value is obtained from CONTROL-M/EM. Reserved for future use.

Figure 142 APPL FORM Parameter Format

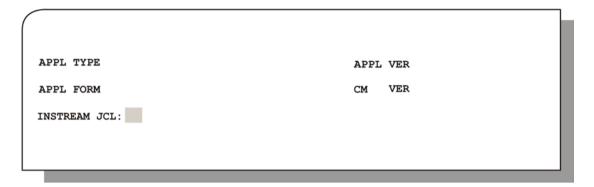


# **APPL TYPE: General Job Parameter**

The application that runs the job.

This is a protected field whose value is obtained from CONTROL-M/EM. Reserved for future use.

Figure 143 APPL TYPE Parameter Format

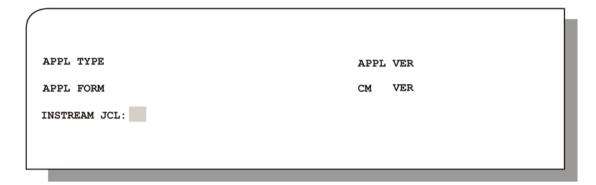


# **APPL VER: General Job Parameter**

The version of the application on which the job runs.

This is a protected field whose value is obtained from CONTROL-M/EM. Reserved for future use.

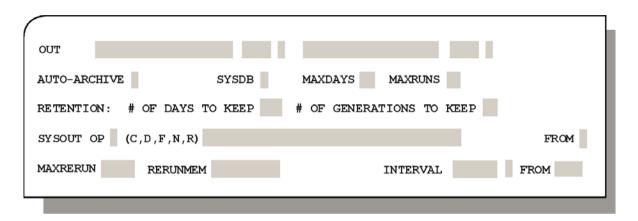
Figure 144 APPL VER Parameter Format



# §Restart§ AUTO-ARCHIVE: Post–Processing Parameter

Controls SYSDATA archiving.

Figure 145 §Restart§ AUTO-ARCHIVE Parameter Format



Optional. The AUTO-ARCHIVE parameter consists of the subparameters described in Table 161.

Table 161 §Restart§ AUTO-ARCHIVE Subparameter Format (part 1 of 2)

Subparameter	Description
AUTO-ARCHIVE	Determines whether SYSDATA is to be archived. Valid values are:
	■ Y (Yes) – Archive SYSDATA. Default.
	<ul> <li>N (No) – Do not archive SYSDATA. If this value is specified for a job, restart of the job under CONTROL-M/Restart and SYSDATA viewing under CONTROL-M is not possible.</li> </ul>
SYSDB	Determines whether all SYSDATA outputs are to be archived to one predesignated data set or whether each SYSDATA output is to be archived to its own data set. Valid values are:
	<ul> <li>Y (Yes) – SYSDATA of all jobs containing a SYSDB value of Y are archived to a common data set. When the common data set is full, another is automatically allocated and used by the system. This is the recommended method. Default.</li> </ul>
	■ N (No) – SYSDATA of each job containing a SYSDB value of N is archived to a unique data set.

Table 161 §Restart§ AUTO-ARCHIVE Subparameter Format (part 2 of 2)

Subparameter	Description
MAXDAYS	Maximum number of days to retain archived SYSDATA value for jobs ended NOTOK. Must be a 2-digit number in the range 00 through 98 or 99. 99 means retain for an unlimited number of days (until deleted by request).
MAXRUNS	Maximum number of runs for which the archived SYSDATA must be retained when the job ended NOTOK. Must be a 3-digit number in the range of 000 through 998 or 999. 999 means retain the SYSDATA of all runs.

#### **General Information**

The AUTO-ARCHIVE subparameter allows you to decide whether to archive SYSDATA, which is defined in "SYSDATA" on page 68. While archiving SYSDATA is normally desirable, it might not be desirable for cyclic jobs, started tasks, or frequently repeated jobs that do not require restart.

If archiving, the SYSDB subparameter allows you to decide whether SYSDATA for different jobs must be archived to a common data set (Y) or whether to use a separate data set for each run (N). If Y is entered, a single archived SYSDATA data set is used for archiving until it is full. Then, another SYSDATA data set is allocated and used. This is the recommended method.

Creating a separate data set for each run is not recommended because:

- Creating many data sets consumes a large amount of space in the disk VTOC.
- Each data set is allocated on a track basis. If the SYSDATA does not completely fill the track, large amounts of disk space may be wasted.

The MAXDAYS and MAXRUNS subparameters define retention criteria for the archived SYSDATA of jobs that ended NOTOK. Defaults are defined in the CONTROL-M/Restart installation parameters. You can specify either or both parameters to override the defaults. If both parameters are specified, retention is limited by the condition that is fulfilled first.

When archiving SYSDATA, BMC Software recommends that you do not enter the value 99 in the MAXWAIT parameter for cyclic jobs or started tasks. Otherwise, these jobs, which are never automatically deleted from the Active Jobs file, may cause the disk to fill up with unnecessary archived SYSDATA. The MAXWAIT parameter is described in "MAXWAIT: Basic Scheduling Parameter" on page 519.

#### NOTE



Specified parameters take effect only during execution of the New Day procedure (CONTDAY) or the CTMCAJF utility. Therefore, it is possible to find more generations of the same job than the current value of MAXRUNS.

### **AUTO-ARCHIVE** and the History File

SYSDATA is archived in the History Jobs file, as defined by the values set for the MAXDAYS and MAXRUNS parameters, and under the following conditions:

As long as a job exists either in the AJF or in the History Jobs file, the action of the MAXDAYS and MAXRUNS parameters applies to the job's SYSDATA, except for the last archived SYSDATA, which will be retained regardless of the setting of the MAXDAYS and MAXRUNS parameters. On the other hand, if the HIST parameter in the CTMPARM member in the IOA PARM library is set to N (No) and a job is deleted from the AJF, the job's SYSDATA is deleted regardless of the values set for the MAXDAYS and MAXRUNS parameters. For more information, see information about maintaining previous runs in the History Jobs file in the *CONTROL-M/Restart User Guide*.

# **Example**

Archive the SYSDATA to a common data set. Retain the archived SYSDATA for 7 days or 20 runs, whichever occurs first.

Figure 146 §Restart§ AUTO-ARCHIVE Parameter Example

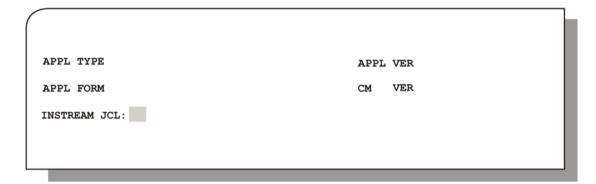
ON PGMST PROCST CODES A/O DO HOUT WHEN TIME + DAYS TO URGN MS				MAXDAYS 07 MAX	
MAXRERUN RERUNMEM INTERVAL FROM STEP RANGE FR (PGM.PROC) . TO .  DN PGMST PROCST CODES A/O  DO  HOUT WHEN TIME + DAYS TO URGN  MS		" -	KEEP 030 # 0F (	GENERATIONS TO K	
STEP RANGE FR (PGM.PROC) . TO . ON PGMST PROCST CODES A/O DO HOUT WHEN TIME + DAYS TO URGN MS				INTEDVA	
ON PGMST PROCST CODES A/O DO SHOUT WHEN TIME + DAYS TO URGN			A DDOC)		_ FRUM
MS	ON PGMST	• • • • • • • • • • • • • • • • • • • •			A/0
	HOUT WHEN	TIME	+ DAY	S TO	URGN
		>>>>> [	END OF SCHEDULI	NG PARAMETERS <<	······································

# **CM VER: General Job Parameter**

The version number of the Control Module (CM) that is used to run the job.

This is a protected field whose value is obtained from CONTROL-M/EM. Reserved for future use.

Figure 147 CM VER Parameter Format

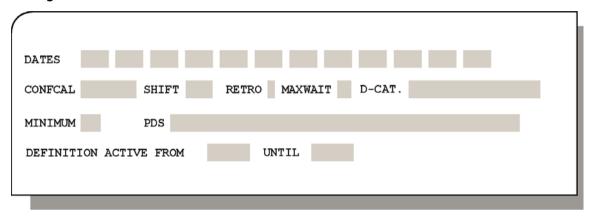


# **CONFCAL: Basic Scheduling Parameter**

Specifies the name of a calendar that is used to confirm the scheduling of the job.

Related parameters are "DAYS: Basic Scheduling Parameter" on page 420, "WDAYS: Basic Scheduling Parameter" on page 669, and "DATES: Basic Scheduling Parameter" on page 418.

Figure 148 CONFCAL Parameter Format



Optional. CONFCAL subparameters are described in Table 162.

Table 162 Optional CONFCAL Subparameters (part 1 of 2)

Subparameter	Description
CONFCAL	A valid 1 through 8 character calendar (member) name. This calendar is used for:
	<ul><li>validating scheduling dates</li><li>determining the scheduled work day</li></ul>
	Jobs to be scheduled on a day, based on other specified Basic Scheduling criteria, are checked against the CONFCAL calendar, as follows:
	■ If the day is a working day in the CONFCAL calendar, the job is tentatively scheduled on that day. This day is referred to as the original scheduling date. Actual scheduling of the job is then determined by the value entered in the SHIFT subparameter (described in this table).
	■ If the day is not a working day in the CONFCAL calendar, the SHIFT subparameter is checked. Depending on the SHIFT value, the job may be scheduled on an earlier or later day, may be scheduled on that day, or may be cancelled.

Table 162 Optional CONFCAL Subparameters (part 2 of 2)

Subparameter	Description
SHIFT	Determines when and if the job must be scheduled. Optional.
	The format of the SHIFT subparameter is xyyy, where
	■ <i>x</i> indicates how to shift scheduling of the job if the original scheduling day of the job is not a working day in the CONFCAL calendar. Valid values are:
	<ul> <li>- ' ' (Blank) - No shifting occurs. The job is not scheduled. Default.</li> <li>-&gt; - Job scheduling is shifted to the next working day in the CONFCAL calendar. Additional shifting may be performed, depending on the <i>yyy</i> value, described below.</li> <li>- '&lt; - Job scheduling is shifted to the previous working day in the CONFCAL calendar. Additional shifting may or may not be performed, depending on the <i>yyy</i> value, described below.</li> <li>- @ - Tentatively schedule the job for the current day, even if the current day is not a working day. Additional shifting may or may not be performed, depending on the <i>yyy</i> value, described below.</li> </ul>
	yyy shifts scheduling of the job forward or backward the specified number of working days, as defined in the CONFCAL calendar. Valid values are:
	— ' ' (Blank) – Do not shift job scheduling any additional time. Default.
	<ul> <li>+nn - Shift job scheduling forward to next nth working day,</li> <li>where n can be as many as 62 working days in the future.</li> </ul>
	— $-nn$ – Shift job scheduling backward to the previous $n$ th working day, where $n$ can be as many as 62 working days in the past.
	yyy does not cause 'negative' scheduling days (-n, -Dn, -Ln, etc.) to be shifted.
	<b>Note:</b> For more information on the use of the SHIFT subparameter, see "The SHIFT Subparameter" below.

# **General Information**

CONFCAL calendars are useful for handling holidays and other scheduling exceptions.

CONFCAL is optional. If no value is set for the CONFCAL parameter, jobs are scheduled according to other basic scheduling criteria without confirmation.

CONFCAL must not contain the name of a periodic calendar. If it does, no day can pass the confirmation.

CONFCAL cannot be used with the PDS and MINIMUM parameters.

#### **The SHIFT Subparameter**

If no CONFCAL calendar is specified, no value can be entered in the SHIFT subparameter, and this field has no effect on job scheduling.

The format and valid values of the SHIFT subparameter are described in Table 162.

The interaction between the *x* value and the *yyy* value of the SHIFT subparameter is as follows:

- If the original scheduling day of the job is a working day in the CONFCAL calendar, the *x* value is ignored and the *yyy* value determines when the job is scheduled.
- If the original scheduling day of the job is not a working day in the CONFCAL calendar, job scheduling is shifted according to the *x* value and then shifted again according to the *yyy* value (if specified) to determine when the job is scheduled.

If the original scheduling day is not a working day and the *x* value is blank, the job is not scheduled (regardless of whether a *yyy* value is specified).

If the result of shifting by *yyy* days is a day that is not allowed, that is, if -n was entered for that day in the DAYS parameter of the job scheduling definition, the job is shifted again to the next allowed working day (for a forward shift) or to the previous allowed working day (for a backward shift).

#### — NOTE —



Prior to version 5.1.4, the SHIFT subparameter consisted of only the x value. If the yyy value is not specified, CONFCAL and SHIFT work as they did prior to version 5.1.4, and version 5.1.4 job scheduling definitions do not need to be changed.

# **Example 1**

This example is based on the following assumptions:

- The current month is September 2001.
- Working days are defined in calendar WORKDAYS, which contains the following working days (indicated by Y) for September 2001:

■ Start of the week is defined as Monday. Weeks start on the following dates in September: 3rd, 10th, 17th, and 24th.

Schedule the rule on the 1st, 7th and 15th day of the month if they are both Saturdays and working days in WORKDAYS. If the day of the month (1st, 7th, 15th) is not a Saturday, do not schedule the rule. If the day of the month is a Saturday but is not a working day, schedule the rule on the next working day.

```
DAYS - 1,7,15
AND/OR - AND
WDAYS - 6
CONFCAL - WORKDAYS
SHIFT - >
```

The rule is scheduled on the days of the month indicated by an asterisk:

Figure 149 Days When Job Scheduled

01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
SA	SU	МО	TU	WE	H	FR	SA	SU	МО	TU	WE	TH	FR	SA	SU	МО	TU	WE	TH	FR	SA	SU	МО	TU	WE	TH	FR	SA	SU
			*													*													

# **Example 2**

Assume that the IOA installation parameter SWEEK was set to SUN (meaning that Sunday is the first working day of the week.)

Schedule the job on every Thursday, except when Friday is a holiday. If Friday is a holiday, schedule the job on the working day prior to Thursday.

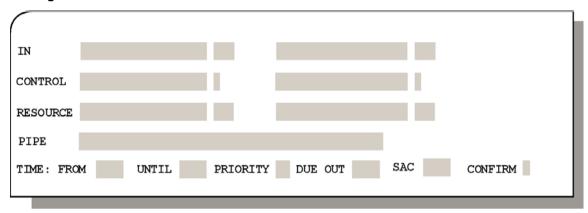
Set the following parameter values:

```
WDAYS 6
CONFCAL WORKDAYS
SHIFT <-01
```

# **CONFIRM: Runtime Scheduling Parameter**

Ensures manual confirmation before a job is submitted or a group is made active.

Figure 150 CONFIRM Parameter Format



Optional. Valid values are described in Table 163.

**Table 163 Optional CONFIRM Parameter Values** 

Parameter	Description
Y (Yes)	Confirmation required. The job is not submitted unless manual confirmation is entered in the Active Environment screen.
N (No)	No confirmation required. The job can be automatically submitted by CONTROL-M without manual confirmation. Default.

## **General Information**

If CONFIRM is set to Y, the job appears in the Active Environment screen with a WAIT CONFIRMATION (FOR SCHEDULE) status. Option C (Confirm) must then be entered in the Active Environment screen for the job to be submitted. When the job is confirmed in the Active Environment screen, the CONFIRM value in the Zoom screen changes to N. If CONFIRM is set to N or left blank, the job is automatically submitted by CONTROL-M at the first available opportunity.

#### NOTE -



In the case of cyclic jobs, confirmation applies to the first run only. Once confirmed, the job is recycled without waiting for subsequent confirmation.

# **Example**

Job OPERCOMP requires manual confirmation in order to be eligible for submission. Manual confirmation can be provided from the Active Environment screen once the job is displayed with a status of WAIT CONFIRMATION (FOR SCHEDULE).

Figure 151 CONFIRM Parameter Example

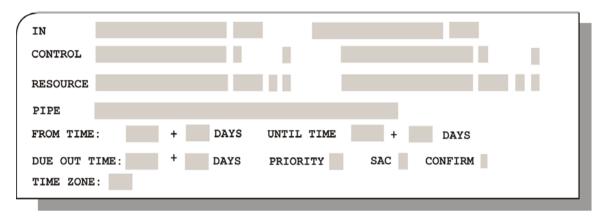
```
JOB: OPERCOMP LIB CTM.PROD.SCHEDULE
                                                      TABLE: OPER
COMMAND ===>
                                                      SCROLL===> CRSR
+-----+
 SCHENV
                          SYSTEM ID
                                                 NJE NODE
 SET VAR
 SET VAR
CTB STEP AT NAME
                               TYPE
 DOCMEM OPERCOMP DOCLIB CTM.PROD.DOC
  DAYS
                                                         AND/OR
  WDAYS
                                                      WCAL
 MONTHS 1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- Y 11- Y 12- Y
 DATES
  CONFCAL
               SHIFT RETRO Y MAXWAIT OO D-CAT
  MINIMUM
               PDS
  DEFINITION ACTIVE FROM
                            UNTIL
  CONTROL
  RESOURCE INIT
                           0001
  PIPE
 FROM TIME + DAYS UNTIL TIME + DAYS
DUE OUT TIME + DAYS PRIORITY SAC CONFIRM Y
  TIME ZONE:
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
                                                           11.17.00
```

1

# **CONTROL: Runtime Scheduling Parameter**

Ensures exclusive or shared control over runtime resources.

Figure 152 CONTROL Parameter Format



Optional. A maximum of two resources can be specified in each CONTROL line. Upon specifying the second resource in a CONTROL line and pressing **Enter**, a new line is opened (for specifying additional resources).

Each CONTROL specification consists of the mandatory subparameters described in Table 164 and may include the optional subparameter described in Table 165.

**Table 164 Mandatory CONTROL Subparameters** 

Subparameter	Description
res-name	User-supplied, descriptive name of 1 through 20 characters used to identify the resource.
state	<ul> <li>Type of control the job requires of the resource. Valid values are:</li> <li>■ E – The job requires exclusive control of the resource during processing.</li> <li>■ S – The job requires shared control of the resource during processing.</li> </ul>

#### **NOTE**



Do not specify the same Control resource in both Exclusive (E) and Shared (S) states in the same job scheduling definition or Group entity, or the job or group cannot run.

In addition, if the resource is in Exclusive state in the Group entity, it must not be specified in any of the jobs belonging to the group; if the resource is in Shared state in the Group, it must not be in Exclusive state in any of the jobs belonging to the group.

**Table 165 Optional CONTROL Subparameter** 

Subparameter	Description
onFail	Whether to keep the Control resource allocated to the job if the job does not end OK. Valid values are:
	■ ' '(blank) – The resource is not kept allocated to the job. Default.
	<ul> <li>K - The resource is kept allocated to the job until one of the following events occurs:         <ul> <li>the job is rerun and ends OK</li> <li>the job is deleted</li> <li>the job is FORCEd OK</li> </ul> </li> </ul>

#### **Example**

CONTROL DISK-VS0020 E K

If the job ends NOTOK, the job continues to hold this exclusive resource.

#### **General Information**

The CONTROL parameter is used to control parallel execution of jobs (and/or groups).

If a job requires a resource in exclusive state, it cannot share usage of that resource with another job (that is, the jobs cannot run in parallel). For example:

- If JOBA requires exclusive control of a resource that is already in use by a different job, JOBA must wait until the other job frees the resource regardless of whether the other job is using the resource in shared or exclusive state.
- If JOBA already has exclusive control of a resource, any job requiring that resource must wait until JOBA frees the resource, regardless of whether the job requires the resource in shared or exclusive state.

If a job requires a resource in shared state, that job can run in parallel with other jobs requiring the same resource in shared state. For example:

- If JOBA requires shared control of a resource that is already in shared use by different jobs, JOBA can use that resource at the same time.
- If JOBA already has shared control of a resource, any job requiring that same resource in shared state can use that resource at the same time.

#### However:

- If JOBA requires shared control of a resource that is already in exclusive use by a different job, JOBA must wait until the other job frees the resource.
- If JOBA already has shared control of a resource, any job requiring that same resource in exclusive state must wait until JOBA frees the resource.

For more information, see "Quantitative and Control Resources" on page 73.

# **Example**

The following three screens (job scheduling definitions) indicate how the CONTROL parameter can control resource usage. All three job scheduling definitions require resource (disk) DISK-VS0020:

- The first job, BKPVS020, is a backup job that requires exclusive control of disk DISK-VS0020.
- The other two jobs, CMPRSJOB and CMPRSSRC, are both compress jobs. They do not require exclusive control (that is, they can share control) of disk DISK-VS0020.

The result is as follows:

- The CMPRSJOB and CMPRSSRC jobs can be run in parallel with each other, but neither can run in parallel with the BKPUS020 job.
- If the BKPVS020 job is running, the CMPRSJOB and CMPRSSRC jobs must wait.
- If either the CMPRSJOB job or the CMPRSSRC job is running, the BKPVS020 job must wait.

This is the first of the three jobs in the example (job BKPVS020).

Figure 153 CONTROL Parameter Example 1

```
JOB: BKPVSO20 LIB CTM.PROD.SCHEDULE
                                                          TABLE: BACKUP
COMMAND ===>
                                                          SCROLL ===> CRSR
 OVERLIB
                                                    STAT CAL
 SCHENV
                            SYSTEM ID
                                                    NJE NODE
  SET VAR
 CTB STEP AT NAME
                                 TYPE
  DOCMEM BKPVS020 DOCLIB CTM.PROD.DOC
                                                         DCAL
  DAYS
                                                             AND/OR
  WDAYS 3.0
                                                         WCAL
  MONTHS 1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- Y 11- Y 12- Y
  DATES
                 SHIFT RETRO Y MAXWAIT OO D-CAT
  CONFCAL
                 PDS
  MINIMUM
  DEFINITION ACTIVE FROM UNTIL
  CONTROL DISK-VS0020 E
  RESOURCE
 FROM TIME + DAYS UNTIL TIME + DAYS
DUE OUT TIME + DAYS PRIORITY SAC CONFIRM
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
                                                               11.17.00
```

This is the second of the three jobs in the example (job CMPRSSRC).

Figure 154 CONTROL Parameter Example 2

```
JOB: CMPRSSCR LIB CTM.PROD.SCHEDULE
                                                        TABLE: OPMAINT
COMMAND ===>
                                                        SCROLL===> CRSR
 OVERLIB
                                                   STAT CAL
                           SYSTEM ID
  SCHENV
                                                   NJE NODE
  SET VAR
  CTB STEP AT NAME
  DOCMEM CMPRSSCR DOCLIB CTM.PROD.DOC
                                                        DCAL
  DAYS
                                                            AND/OR
  WDAYS
                                                        WCAL
  MONTHS 1- 2- 3- 4- 5- 6- 7- 8- 9- 10- 11- 12-
               SHIFT RETRO Y MAXWAIT OO D-CAT
 MINIMUM 025 PDS GSD.DEPO.SCR
  DEFINITION ACTIVE FROM UNTIL
  ΤN
  CONTROL DISK-VS0020 S
  RESOURCE
  FROM TIME + DAYS UNTIL TIME + DAYS
DUE OUT TIME + DAYS PRIORITY SAC CONFIRM
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
                                                              11.17.00
```

This is the third of three jobs in the example (job CMPRSJOB).

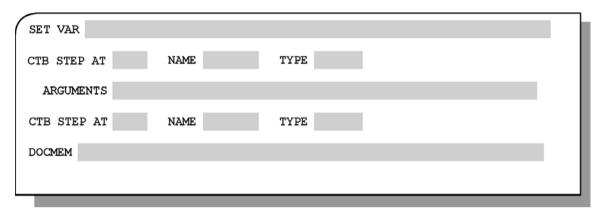
Figure 155 CONTROL Parameter Example 3

```
JOB: CMPRSJOB LIB CTM.PROD.SCHEDULE
                                                         TABLE: OPMAINT
COMMAND ===>
                                                         SCROLL==> CRSR
  OVERLIB
                                                   STAT CAL
                            SYSTEM ID
                                                   NJE NODE
  SET VAR
  CTB STEP AT NAME
                                 TYPE
  DOCMEM CMPRSJOB DOCLIB CTM.PROD.DOC
  DAYS
                                                            AND/OR
  WDAYS
                                                         WCAL
  MONTHS 1- 2- 3- 4- 5- 6- 7- 8- 9- 10- 11- 12-
  DATES
  CONFCAL
                SHIFT
                           RETRO Y MAXWAIT OO D-CAT
  CONFCAL SHIFT RETRO Y
MINIMUM 020 PDS GSD.DEPO.JOB
  DEFINITION ACTIVE FROM UNTIL
  ΙN
  CONTROL DISK-VS0020
  RESOURCE
  FROM TIME + DAYS
DUE OUT TIME + DAYS
                              UNTIL TIME +
                                                 DAYS
                              PRIORITY SAC CONFIRM
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
                                                                11.17.00
```

# **CTB STEP: General Job Parameter**

Adds CONTROL-M/Analyzer steps as the first and/or last step of the execution of the job.

Figure 156 CTB STEP Parameter Format



Optional. The CTB STEP parameter consists of the subparameters described in Table 166.

**Table 166 Optional CTB STEP Parameters** 

Parameter	Description
AT	Indicates where to place the CONTROL-M/Analyzer step in the job. Mandatory. Valid values are:
	<ul> <li>START – The indicated CONTROL-M/Analyzer step must become the first step of the job.</li> <li>END – The indicated CONTROL-M/Analyzer step must become the last step of the job.</li> </ul>
NAME	Name of the CONTROL-M/Analyzer entity. Must be a valid name of a CONTROL-M/Analyzer rule or mission. Mandatory.
ТҮРЕ	Type of CONTROL-M/Analyzer entity. Mandatory. Valid values are:  RULE – Entity is a CONTROL-M/Analyzer rule.
	■ MISSION – Entity is a CONTROL-M/Analyzer mission.
ARGUMENTS	Arguments to be passed to the CONTROL-M/Analyzer step. Optional.

#### - NOTE -



The ARGUMENTS line is not displayed until the CTB STEP line is filled in and **Enter** is pressed.

# **General Information**

A maximum of two CTB STEP statements (that is, one START statement and one END statement) can be entered.

Upon filling in the first CTB STEP line on the screen and pressing **Enter**, the ARGUMENTS line and the second CTB STEP line are displayed. If the second CTB STEP line is filled in and **Enter** is pressed, its ARGUMENTS line is displayed.

Multiple arguments must be separated by a comma without a space because they are automatically passed to the CONTROL-M/Analyzer step as a PARM='arguments' parameter in the JCL of the step.

CONTROL-M uses the status returned by CONTROL-M/Analyzer as it would use the return status of any job step.

- If CONTROL-M/Analyzer returns a status of OK or TOLER (within accepted tolerances), CONTROL-M considers the step as having ended OK.
- If CONTROL-M/Analyzer returns a status of NOTOK or ABEND, CONTROL-M considers the job step as having ended NOTOK.

# **Example**

After successfully performing salary calculations, job SACALC01 invokes rule CHKCALC to ensure that the results are reasonable, and then sets OUT condition SALARY-OK.

#### Figure 157 CTB STEP Parameter Example

```
JOB: SACALCO1 LIB CTM.PROD.SCHEDULE
                                                         TABLE: SALARY
COMMAND ===>
                                                         SCROLL===> CRSR
  MEMNAME SACALCO1 MEMLIB GENERAL
  OWNER SYS1 TASKTYPE JOB PREVENT-NCT2 DFLT N
 APPL SAL
DESC SALARY CALCULATIONS
                                 GROUP SALARY
  OVERLIB
                                                    STAT CAL
  SCHENV
                       SYSTEM ID
                                                   NJE NODE
  SET VAR
  CTB STEP AT END NAME CHKCALC TYPE RULE
   ARGUMENTS %%ODATE
  CTB STEP AT NAME TYPE DOCMEM SACALCO1 DOCLIB CTM.PROD.DOC
                   NAME
  DAYS 01,15
                                                          DCAL
                                                          AND/OR
  WDAYS
                                                          WCAL
  MONTHS 1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- Y 11- Y 12- Y
  CONFCAL
                 SHIFT RETRO Y MAXWAIT 00 D-CAT
  MINIMUM
                 PDS
  DEFINITION ACTIVE FROM UNTIL
  ΤN
  CONTROL
  RESOURCE
  PIPE
  FROM TIME + DAYS
DUE OUT TIME + DAYS
                              UNTIL TIME
                              PRIORITY SAC CONFIRM
  TIME ZONE:
          SALARY-OK ODAT +
  OUT
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
                                                             11.17.00
```

# **D-CAT: Basic Scheduling Parameter**

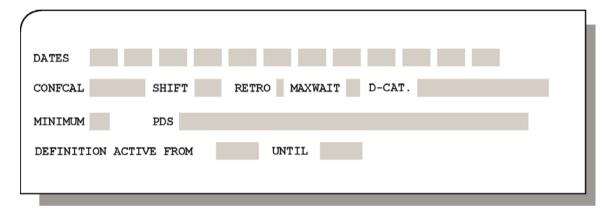
Name of a CONTROL-D report decollating mission category that must be scheduled under CONTROL-D when the job is scheduled under CONTROL-M.

#### - NOTE



Prior to version 5.1.4, the D-CAT parameter was called CATEGORY.

Figure 158 DCAT Parameter Format



Optional. The D-CAT parameter must be 1 through 20 characters, or * for all categories.

If this parameter is specified when CONTROL-D is not installed, New Day processing stops immediately after this job. For a description of New Day processing, see Chapter 6, "Selected Implementation Issues," and the *INCONTROL* for z/OS Administrator Guide.

### **General Information**

If the parameter is specified, whenever the job is scheduled, a search is made in the CONTROL-D report decollating mission library for a job (member) with the name entered in the MEMNAME parameter, which is described in "MEMNAME: General Job Parameter" on page 528, and with the same category. (No search is made in the case of job restarts.)

Generally, the selected category is forced and placed in the CONTROL-D Active Missions file (that is, the output of the job must be decollated by CONTROL-D). If D-CAT is set to *, all categories of the job are forced under CONTROL-D.

#### NOTE -



If the CTGFORC parameter of the CTMPARM member in the IOA PARM library is set to NO, selected categories are scheduled (that is, not forced).

# **Example**

The job output must be decollated by the CONTROL-D report decollating mission category DAILY.

#### Figure 159 DCAT Parameter Example

```
JOB: GNRLDR12 LIB CTM.PROD.SCHEDULE
                                                                      TABLE: GNRLDR
COMMAND ===>
                                                                      SCROLL===> CRSR
 MEMNAME GNRLDR12 MEMLIB GENERAL

OWNER SYS1 TASKTYPE JOB PREVENT-NCT2 DFLT N

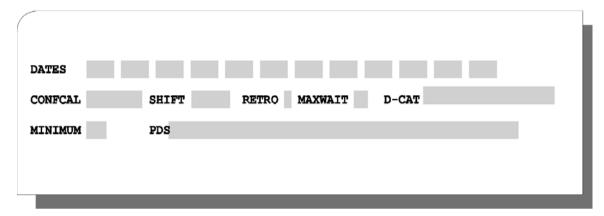
APPL GENERAL GROUP GENERAL-LEDGER

DESC GENERAL IFDGER DAILY REPORTS
  DESC
          GENERAL LEDGER DAILY REPORTS
  OVERLIB
                                                                STAT CAL
                                    SYSTEM ID
  SCHENV
                                                                NJE NODE
  SET VAR
  CTB STEP AT
                     NAME
                                         TYPE
  DOCMEM GNRLDR12 DOCLIB CTM.PROD.DOC
  DAYS
           ALL
                                                                       DCAL WORKDAYS
                                                                            AND/OR
  WDAYS
  MONTHS 1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- Y 11- Y 12- Y
  DATES
                                  RETRO Y MAXWAIT 00 D-CAT DAILY
  CONFCAL
                     SHIFT
  MINIMUM
                    PDS
  DEFINITION ACTIVE FROM
                                     UNTIL
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
                                                                              11.17.00
```

# **DATES: Basic Scheduling Parameter**

Specifies dates, by month and day, on which the job must be scheduled for execution.

Figure 160 DATES Parameter Format



Optional. The DATES parameter specifies a valid 4-character date, in either mmdd or ddmm format, depending on site format.

A maximum of 12 dates can be specified.

### **General Information**

The job is scheduled for execution only on the dates specified in the DATES parameter.

The DATES parameter cannot be used with the PDS, MINIMUM, DAYS, and DCAL parameters.

If values are set for both the MONTHS parameter and the DATES parameter, the MONTHS parameter setting is ignored.

To specify more than 12 dates for one job, define the dates in a calendar (instead of using this parameter) and specify the calendar in the DCAL (or WCAL) subparameter.

As an alternative to using calendars for specifying more than twelve dates in jobs belonging to a group, up to twelve dates can be specified in a Schedule Tag definition in the Group entity, and multiple schedule tags of this type can be defined. These can then be specified in the jobs.

The relationship between DATES and WDAYS and WCAL is OR. If the job is to be scheduled according to the DATES parameter or according to the WDAYS and WCAL combination, it is scheduled.

# **Examples**

### **Example 1**

Schedule a job for the 15th of January (mmdd format).

DATES 0115

#### **Example 2**

Schedule job PRDKPL01 for the 21st of June and the 21st of December (ddmm format).

#### Figure 161 DATES Parameter Example

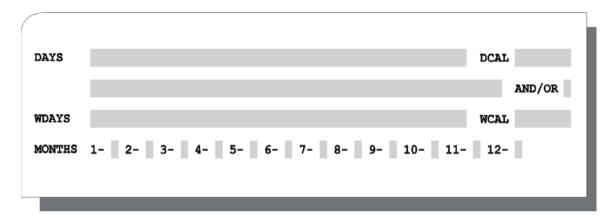
```
JOB: PRDKPL01 LIB CTM.PROD.SCHEDULE
                                                            TABLE: PRODKPL
COMMAND ===>
                                                            SCROLL===> CRSR
  MEMNAME PRDKPL01 MEMLIB CTM.PROD.JCL
  OWNER SYS1 TASKTYPE JOB PREVENT-NCT2 DFLT N
                                    GROUP PROD-KPL
         DAILY PRODUCTION - START OF APPL-PROD-KPL
  DESC
  OVERLIB
                                                       STAT CAL
  SCHENV
                               SYSTEM ID
                                                       NJE NODE
  SET VAR
  CTB STEP AT
                    NAME
                                    TYPE
  DOCMEM PRDKPL01
                    DOCLIB CTM.PROD.DOC
                                                             DCAL
  DAYS
                                                                 AND/OR
                                                             WCAL
  MONTHS 1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- Y 11- Y 12- Y
  DATES 2106 2112
  CONFCAL
                  SHIFT
                             RETRO Y MAXWAIT OO D-CAT
  MINIMUM
                  PDS
  DEFINITION ACTIVE FROM
                                UNTIL
          START-DAILY-PROD-KPL ODAT
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
                                                                  11.17.00
```

# **DAYS: Basic Scheduling Parameter**

The days of the month on which the job must be scheduled.

Related parameters are "WDAYS: Basic Scheduling Parameter" on page 669 and "CONFCAL: Basic Scheduling Parameter" on page 402.

Figure 162 DAYS Parameter Format



Optional. The DAYS parameter specifies days of the month on which the job must be scheduled, provided other basic scheduling criteria are met. Values for DAYS can be specified alone, or they can be specified together with a calendar specified in the DCAL subparameter. DAYS and DCAL can also be specified together with WDAYS and WCAL (described under "WDAYS: Basic Scheduling Parameter" on page 669).

The DAYS subparameters are described in Table 167.

Table 167 DAYS Subparameters (part 1 of 2)

Subparameter	Description
days	Days of the month on which to schedule a job. The months in which to order jobs are specified in the MONTHS parameter, described in "MONTHS: Basic Scheduling Parameter" on page 533. Various formats (described later) can be used to specify DAYS (for example, 2 means the second day of the month, L2 means the day before the last day of the month, D1PA means the first day in period A).

Table 167 DAYS Subparameters (part 2 of 2)

Subparameter	Description
DCAL	Name of a calendar containing a predefined set of dates (referred to as working days) on which a job must be scheduled. A specified value must be either a valid member name of 1 through 8 characters, or an * to indicate that the calendar specified in the CONFCAL parameter must be used for scheduling. For more information on how to define, use and modify calendars, see "IOA Calendar Facility" on page 306.
	Note: A calendar specified in DCAL does not have to exist when defining the DAYS parameter. It must exist when the job is being ordered.
AND/OR	Conjunctional parameter used to link the DAYS and WDAYS parameters when both are specified.  • A (AND) – Both DAYS (/DCAL) and WDAYS (/WCAL) criteria must be met in order for a job to be scheduled.
	<ul> <li>O (OR) - DAYS (/DCAL) and/or WDAYS (/WCAL) criteria must be met for a job to be scheduled. Default.</li> </ul>
	If A (AND) is specified when either DAYS or WDAYS is specified (but not both), the missing DAYS or WDAYS value is automatically set to ALL.

Assuming all other basic scheduling criteria are met:

- When DAYS are specified without DCAL, the job is scheduled on the specified days (in the specified months).
- When DCAL is specified without DAYS, the job is scheduled on all working days marked in the DCAL calendar.
- When DAYS and DCAL are both specified, scheduling depends on how the working days are defined in the calendar and the values or format of the DAYS parameter combine, as described in the following paragraphs.
- When both DAYS and WDAYS criteria are specified, scheduling depends on the AND/OR subparameter connecting them.

### **Valid Formats for DAYS**

Valid formats for the DAYS parameter, and how they relate to DCAL, are described in the following paragraphs.

In the following non-periodic scheduling formats:

- n is an integer from 1 through 31.
- Multiple values can be specified (separated by commas) in any order.
- A calendar name specified for DCAL must not designate a periodic calendar.

**Table 168 Non-Periodic Scheduling Formats** 

Parameter	Description
ALL	All days of the month. If ALL is specified, other DAYS values cannot be specified with it.
	If a DCAL calendar is not defined, schedule the job on all days in the month. If a DCAL calendar is defined, schedule the job only on the working days indicated in the calendar.
n,	Specific days of the month.
	If a DCAL calendar is not defined, schedule the job on the specified days.
	If a DCAL calendar is defined, schedule the job only when a day is defined as a working day in both the DAYS parameter and the DCAL calendar.
+n,	Days of the month in addition to the working days specified in the DCAL calendar. DCAL is mandatory.
-n,	Order the job on all days except the <i>n</i> th day from the beginning of the month. DCAL is mandatory.
>n,	Order the job on the indicated day if it is a working day in the DCAL calendar; otherwise, order the job on the next working day that is not negated by a $-n$ value in this parameter. This format is frequently used for holiday handling. DCAL is mandatory.
<n,< td=""><td>Order the job on the indicated day if it is a working day in the DCAL calendar; otherwise, order the job on the last previous working day of the month that is not negated by a –n value in this parameter. This format is frequently used for holiday handling. DCAL is mandatory.</td></n,<>	Order the job on the indicated day if it is a working day in the DCAL calendar; otherwise, order the job on the last previous working day of the month that is not negated by a –n value in this parameter. This format is frequently used for holiday handling. DCAL is mandatory.
Dn,	Order the job on the <i>n</i> th working day from the beginning of the month. DCAL is mandatory.
-D <i>n</i> ,	Order the job on all working days except the <i>n</i> th working day from the beginning of the month. DCAL is mandatory.
Ln,	Order the job on the <i>n</i> th day (or <i>n</i> th working day if DCAL is defined) counting backward from the end of the month. DCAL is optional.
-Ln,	If DCAL is defined, order the job on all working days except the nth working day counting backward from the end of the month. If DCAL is not defined, order the job on all days except the <i>n</i> th day counting backward from the end of the month. DCAL is optional.

In the following periodic scheduling formats:

- *n* is any integer from 1 through 63, and *i* is any valid period identifier.
- An * can be specified as the *i* value to represent all periods.
- An * can be specified as the n value in format DnPi to represent all days. (* is not a valid n value in formats –DnPi, LnPi and –LnPi.)
- A period can span any number of days, but by default, no more than 33 days can elapse after the appearance of one identifier in a period until the appearance of the next matching identifier in the same period. Once a gap of 33 days has been reached, the period automatically closes. However, the INCONTROL administrator can change the 33-day default by changing the value in member IOADFLT in the IOAENV library.
- The name of a periodic calendar must be specified in DCAL. For details about periodic calendars, see "IOA Calendar Facility" on page 306.

**Table 169 Periodic Scheduling Formats** 

Format	Description
D <i>n</i> P <i>i</i> ,	Order the job on the <i>n</i> th day of period <i>i</i> from the beginning of the period.
-DnPi,	Order the job on all days of period <i>i</i> except the <i>n</i> th day of period <i>i</i> from the beginning of the period.
LnPi,	Order the job on the <i>n</i> th day of period <i>i</i> counting backward from the last day of the period.
-LnPi,	Order the job on all days of period <i>i</i> except the <i>n</i> th day of period <i>i</i> counting backward from the last day of the period.

### **General Information**

Negative values take precedence over positive values when determining whether a job must be scheduled on a certain date. If a negative value (that is, format –n, –Dn, –Ln, –DnPi, or – LnPi) in either the DAYS or WDAYS field prevents a job from being scheduled on a date, the job is not scheduled on that date even if a positive value (such as Ln) in a basic scheduling parameter would otherwise result in the job being scheduled on that date.

A maximum of eight periodic values of type DnPi, –DnPi, LnPi, and/or –LnPi can be designated in any desired order.

If periodic and non-periodic values are mixed when specifying the DAYS parameter, processing depends on the calendar type specified in the DCAL parameter:

- If a non-periodic calendar is specified in the DCAL parameter, only non-periodic values in the DAYS parameter are processed; periodic values are ignored. In this case, negative periodic values (that is, -DnPi, -LnPi) are also ignored and do not supersede other values.
- If a periodic calendar is specified in the DCAL parameter, all periodic values and the negative non-periodic value -n in the DAYS parameter are processed; all other non-periodic values are ignored.

The MONTHS parameter is ignored when periodic values are specified in the DAYS parameter.

For information about certain exceptional situations in the interaction of the DAYS and MONTHS parameters, see "MONTHS: Basic Scheduling Parameter" on page 533.

The DAYS parameter cannot be used with the PDS, MINIMUM, and DATES parameters.

# **Examples**

The examples in this chapter are based on the following assumptions:

- The current month is December, 2001.
- Working days are defined in calendar WORKDAYS, which contains the following working days (indicated by Y) for December, 2001.

- WDAYS are defined as working days beginning on Monday.
- Periodic calendar PERIDAYS contains the following periodic definition for December 2001. These examples assume that all other days of this calendar are blank.

```
1 2 3 4 5 6 7 8 9 + 1 2 3 4 5 6 7 8 9 + 1 2 3 4 5 6 7 8 9 + 1
B C A A B B C A A B B C A A B B C A A B B
```

■ Start of the week is defined as Monday. Weeks start on the following dates in December 2001: 3rd, 10th, 17th, 24th, and 31st.

At the end of each example, asterisks in a December 2001 calendar indicate the days on which the job is scheduled.

#### **Example 1**

Schedule the job on the 17th day and the last day of the month.

DAYS 17,L1

The job is scheduled on the days of the month indicated by an asterisk:

Figure 163 DAYS Parameter Example 1

01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
SA	SU	МО	TU	WE	TH	FR	SA	SU	МО	TU	WE	ТН	FR	SA	SU	МО	TU	WE	TH	FR	SA	SU	МО	TU	WE	TH	FR	SA	su	МО
																*														*

### **Example 2**

Schedule the job on all working days of the month except the 6th day of the month, and also schedule the job on the 1st day of the month.

DAYS +1,-6
DCAL WORKDAYS

The job is scheduled on the days of the month indicated by an asterisk:

Figure 164 DAYS Parameter Example 2

01	02	03	04	05	06	07	80	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
SA	SU	МО	TU	WE	TH	FR	SA	su	МО	TU	WE	ТН	FR	SA	SU	МО	TU	WE	ΤH	FR	SA	SU	MO	TU	WE	TH	FR	SA	su	МО
*		*	*	*		*			*	*	*	*	*			*	*	*	*	*			*		*	*	*			*

#### **Example 3**

Schedule the job on all working days of the month except the first and last working days, and except the 17th day, of the month.

DAYS -D1,-17,-L1
DCAL WORKDAYS

The job is scheduled on the days of the month indicated by an asterisk:

Figure 165 DAYS Parameter Example 3

01	02	03	04	05	06	07	80	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
SA	SU	МО	TU	WE	TH	FR	SA	su	МО	TU	WE	ТН	FR	SA	SU	МО	TU	WE	ΤH	FR	SA	SU	MO	TU	WE	TH	FR	SA	su	МО
			*	*	*	*			*	*	*	*	*				*	*	*	*			*		*	*	*			

#### **Example 4**

Schedule the job on the 8th day of the month. If it is not a working day, schedule the job on the closest preceding working day.

DAYS	<8		
DCAL	WORKDAYS		

The job is scheduled on the days of the month indicated by an asterisk:

Figure 166 DAYS Parameter Example 4

01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
SA	SU	МО	TU	WE	ТН	FR	SA	su	МО	TU	WE	TH	FR	SA	SU	МО	TU	WE	크	FR	SA	SU	МО	TU	WE	TH	FR	SA	SU	МО
						*																								

#### **Example 5**

Schedule the job on the 1st day of period A, and on all days, except the 2nd day, of period B. Do not schedule the job on the 5th day of the month.

```
DAYS -5,D1PA,-D2PB
DCAL PERIDAYS
```

The job is scheduled on the days of the month indicated by an asterisk:

Figure 167 DAYS Parameter Example 5

01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
SA	SU	МО	IJ	WE	TH	FR	SA	su	МО	TU	WE	H	FR	SA	SU	МО	IJ	WE	ТН	FR	SA	SU	МО	TU	WE	TH	FR	SA	su	МО
		*							*				*			*				*			*				*			*

### **Example 6**

Schedule the job on each Monday and on the 1st day of the month.

DAYS	1			
AND/OR	OR			
WDAYS	1			

The job is scheduled on the days of the month indicated by an asterisk:

Figure 168 DAYS Parameter Example 6

01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
SA	SU	МО	IJ	WE	ĭ	FR	SA	su	МО	IJ	WE	Ŧ	FR	SA	SU	МО	TU	WE	픋	FR	SA	SU	МО	TU	WE	ΤH	FR	SA	SU	МО
*		*							*							*							*							*

### **Example 7**

Schedule the job on the 3rd day of the month provided it is a Monday.

DAYS	3
AND/OR	AND
WDAYS	1

The job is scheduled on the days of the month indicated by an asterisk:

Figure 169 DAYS Parameter Example 7

01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
SA	SU	МО	TU	WE	ТН	FR	SA	su	МО	TU	WE	ТН	FR	SA	SU	МО	TU	WE	프	FR	SA	SU	МО	TU	WE	TH	FR	SA	su	МО
		*																												

#### **Example 8**

Schedule the job on the last Monday of the month.

DAYS	L1,L2,L3,L4,L5,L6,L7
AND/OR	AND
WDAYS	1

The job is scheduled on the days of the month indicated by an asterisk:

Figure 170 DAYS Parameter Example 8

01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
SA	SU	МО	TU	WE	ТН	FR	SA	SU	МО	TU	WE	ТН	FR	SA	SU	МО	TU	WE	TH	FR	SA	SU	МО	TU	WE	TH	FR	SA	su	МО
																														*

#### **Example 9**

Schedule the job on the 1st, 7th and 15th days of the month if they are both Saturdays and working days. If the day of the month (1st, 7th, 15th) is not a Saturday, do not schedule the job. If the day of the month is a Saturday, but it is not a working day, schedule the job on the next working day.



The job is scheduled on the days of the month indicated by an asterisk:

Figure 171 DAYS Parameter Example 9

01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
SA	SU	МО	TU	WE	ΤH	FR	SA	su	МО	TU	WE	ΤH	FR	SA	SU	МО	TU	WE	TH	FR	SA	SU	МО	TU	WE	ТН	FR	SA	SU	МО
		*														*														

#### **Example 10**

Schedule the job to run on the first Friday after the 15th of the month.

DAYS	16,17,18,19,20,21,22
AND/OR	AND
WDAYS	5

The job is scheduled on the days of the month indicated by an asterisk:

Figure 172 DAYS Parameter Example 10

01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
SA	SU	МО	TU	WE	TH	FR	SA	su	МО	TU	WE	ТН	FR	SA	SU	МО	TU	WE	ΤH	FR	SA	SU	МО	TU	WE	TH	FR	SA	SU	МО
																				*										

### **Example 11**

Schedule the job to run on the 15th day of every period, except for period G. The periods are defined in periodic calendar PERCAL.

The following steps are required:

1 In the group entity of a group scheduling table, define a tag, TAG1, which contains the following scheduling criteria:

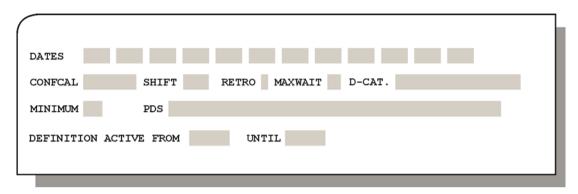
2 In the actual job definition, specify the following:

```
SCHEDULE TAG TAG1
RELATIONSHIP (AND/OR) A
DAYS D15P* DCAL PERCAL
```

# **DEFINITION ACTIVE: Basic Scheduling Parameter**

Specifies the time limits (FROM, UNTIL) for using the job scheduling definition.

Figure 173 DEFINITION ACTIVE Parameter Format



Optional. The parameters include the subparameters described in Table 168.

**Table 170 DEFINITION ACTIVE Subparameters** 

Subparameter	Description
FROM	6-digit date. The job scheduling definition will only be used if the ordering date is later than the date specified. Default: ''(Blank)
UNTIL	6-digit date. The job scheduling definition will only be used if the ordering date is earlier than the date specified. Default: ''(Blank)

The format of either the FROM or UNTIL subparameters may be ddmmyy, mmddyy, or yymmdd, depending on your local site standard, as set by the DATETYP parameter in the IOAPARM member in the IOA PARM library.

## **General Information**

FROM and UNTIL dates together define a time frame for ordering the job. Unless forced, a job can only be ordered during the specified time frame. However, if the job is forced, the FROM and UNTIL parameters are ignored.

■ If you specify both the FROM and UNTIL subparameters for a particular job, the job can only be ordered on or later than the date specified in the FROM subparameter, and on or earlier than the date specified in the UNTIL subparameter. There are two possibilities:

1. The date specified in the FROM subparameter is earlier than that specified in the UNTIL subparameter.

For example,

DEFINITION ACTIVE FROM 091002 UNTIL 011102

The job can only be ordered on or between October 9, 2002 and November 1, 2002.

2. The date specified in the FROM subparameter is later than that specified in the UNTIL subparameter.

For example,

DEFINITION ACTIVE FROM 090502 UNTIL 010402

The job can only be ordered on or after May 9, 2002, or before or on April 1, 2002, but not between those dates.

■ If you specify the FROM subparameter, but not the UNTIL subparameter, the job cannot be ordered before the date specified, but can be ordered on or at any time later than that date.

For example,

DEFINITION ACTIVE FROM 091002 UNTIL

The job can only be ordered on or after October 9, 2002.

■ If you do not specify the FROM subparameter, but specify the UNTIL subparameter, the job can be ordered on or at any time earlier than the date specified, but not later than that date.

For example,

DEFINITION ACTIVE FROM

UNTIL 011102

The job can only be ordered on or before November 1, 2002.

■ If you do not specify either the FROM or UNTIL subparameters, there is no restriction on the date when the job can be ordered.

For example,

DEFINITION ACTIVE FROM

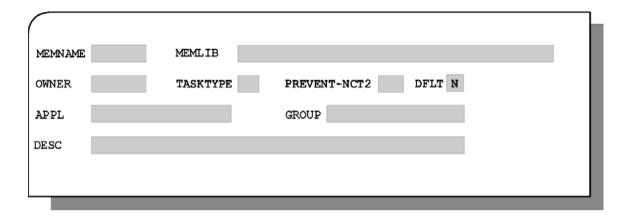
UNTIL

The job can be ordered on any date.

# **DESC: General Job Parameter**

Description of the job to be displayed in the Job List screen.

Figure 174 DESC Parameter Format



Optional. This parameter may consist of text of 1 through 50 characters.

### **General Information**

The DESC parameter is informational. It does not affect job processing. It serves as internal documentation, to let the user know the purpose of (or some other key information about) the job.

The description specified in the DESC parameter appears to the right of the job name in the Job List screen.

Text for the DESC parameter can be specified in any language.

#### NOTE



For conversion customers prior to version 6.0.00, if the current job was converted from another job scheduling product, such as CA-7, the string SCHEDULE-PREV-DAY or SCHEDULE-PREV-ONLY may appear in the DESC field for the job group. This string causes all scheduled runs of the job to be shifted back one day. The SAC parameter is used instead.

For information on how to specify more detailed job documentation, see "Job Documentation" on page 144.

# **Example**

Job OPERCOMP appears in the Job List screen with the description: JOB RUN ON THE 1ST OF THE MONTH.

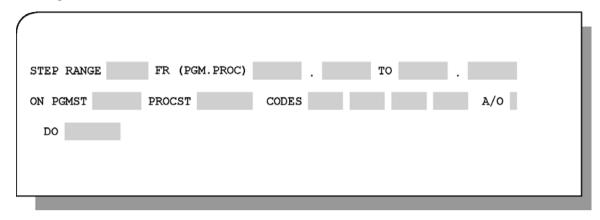
#### Figure 175 DESC Parameter Example

```
JOB: OPERCOMP LIB CTM.PROD.SCHEDULE
                                                         TABLE: OPER
                                                          SCROLL===> CRSR
COMMAND ===>
  MEMNAME OPERCOMP MEMLIB CTM.PROD.JCL
  OWNER SYS1 TASKTYPE JOB PREVENT-NCT2 DFLT N
  APPL OPER
                                  GROUP MAINTENANCE
  DESC JOB RUN ON THE 1ST OF THE MONTH
                                                       STAT CAL
  OVERLIB
  SCHENV
                            SYSTEM ID
                                                       NJE NODE
  SET VAR
  CTB STEP AT NAME TYPE
DOCMEM OPERCOMP DOCLIB CTM.PROD.DOC
  DAYS 01
                                                          DCAL
                                                              AND/OR
                                                          WCAL
  WDAYS
  MONTHS 1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- Y 11- Y 12- Y
  DATES
  CONFCAL
                 SHIFT RETRO Y MAXWAIT 00 D-CAT
  MINIMUM
                 PDS
  DEFINITION ACTIVE FROM
                             UNTIL
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
```

# **DO Statement: Post-Processing Parameter**

Actions taken when the ON step and codes event criteria are satisfied.

Figure 176 DO Parameter Format



Optional. Specify DO statements as follows:

- Type the action keyword (such as COND) in the DO field and press **Enter**.
- In many cases, subparameter fields are displayed. Fill in the subparameters and press **Enter** again.
- Multiple DO statements can be specified. After entering a DO statement, another DO line is automatically displayed.

DO actions are described in Table 171. Each is discussed in detail in this chapter.

Table 171 DO Actions (part 1 of 2)

DO Action	Description
DO COND	Adds and/or deletes a prerequisite condition.
DO CTBRULE	Invokes a CONTROL-M/Analyzer rule.
DO FORCEJOB	Forces a job order under CONTROL-M.
§Restart§ DO IFRERUN	If a rerun is necessary for the job, specifies restart parameters for CONTROL-M/Restart.
DO MAIL	Sends an e-mail to the specified recipients.
DO NOTOK	Sets the job step status to NOTOK.
DO OK	Sets the job step status to OK.
DO REMEDY	Open a Remedy Helpdesk ticket.
DO RERUN	Reschedules the job (for rerun).
DO SET	Assigns a value to an AutoEdit variable.
DO SHOUT	Sends a message to a specified destination.

Table 171 DO Actions (part 2 of 2)

DO Action	Description
DO STOPCYCL	Stops recycling a cyclic task.
DO SYSOUT	Handles the job output.

#### **General Information**

DO statements are generally paired with the preceding ON PGMST, ON PROCST, or ON CODES statements (all of which are described in this chapter). Their implied relationship is:

Table 172 Relationship of DO Statements with Other Statements

Statement	Description
IF	On step and code event criteria (specified in the ON PGMST, ON PROCST, or ON CODES statements) are satisfied.
THEN	Perform all actions specified in the DO statements.

All specified DO statements have an AND relationship.

To add an empty DO statement between two existing DO statements, type the > character over the first letter in the DO field of the earlier DO statement, and press **Enter**.

### **Example**

DO >OND

>OND is restored to its original value, COND, when **Enter** is pressed (the > character disappears).

To delete unwanted DO statements, either delete the DO keyword and press **Enter** or specify appropriate Line Editing commands in the Edit environment, which is described in Appendix A, "The CONTROL-M Application Program Interface (CTMAPI),"

# **DO COND: Post-Processing Parameter**

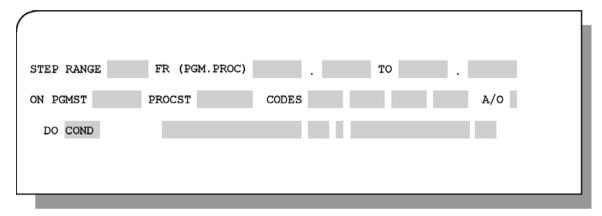
Specifies prerequisite conditions to be added or deleted if the accompanying ON step and code criteria are satisfied.

#### **NOTE**



DO COND and OUT statements are similar, but not identical. The differences are outlined in "Differences between OUT and DO COND" on page 440.

Figure 177 DO COND Parameter Format



Optional. Type COND in the DO field and press Enter.

A maximum of two prerequisite conditions can be specified in each standard DO COND line. One prerequisite condition can be specified in each long DO COND line. When you specify the second prerequisite condition in a standard DO COND line, or one prerequisite condition in a long DO COND line, and press **Enter**, a new DO COND line is opened for specifying additional prerequisite conditions. For more information, see "Specifying Long DO COND Condition Names" on page 439.

Each DO COND statement consists of the mandatory subparameters described in Table 173.

Table 173 DO COND Mandatory Subparameter Formats (part 1 of 2)

Subparameter	Description
cond-name	User-supplied descriptive name of 1 through 39 characters, used to identify the condition.
	<b>Note</b> : A condition name must not begin with the symbols " $\mid$ ", " $\neg$ ", or " $\setminus$ ", and must not contain parentheses (), because each of these characters has a special meaning.
	You can use an AutoEdit variable in a condition name, provided that the AutoEdit variable has a value that is known before the job is ordered.
dateref	4-character date reference. Valid values are:
	<ul> <li>date – A specific date, in either mmdd or ddmm format, depending on the site standard</li> </ul>
	■ ODAT – Resolves to the original scheduling date. Default.
	■ +nnn – Resolves at job order time to ODATE+nnn calendar days. nnn is three digits (000-999).
	■ -nnn – Resolves at job order time to ODATE-nnn calendar days. nnn is three digits (000-999).
	<b>Note:</b> +001 and -001 are not necessarily the same as NEXT and PREV, because NEXT and PREV are based on job scheduling criteria, while + <i>nnn</i> and - <i>nnn</i> are based on calendar days.
	■ PREV – Resolves to the previous date on which the job ought to have been scheduled, according to its basic scheduling criteria (or ODATE-1 for a forced job).
	■ NEXT – Resolves to the next date on which the job is scheduled, according to its basic scheduling criteria (or ODATE+1, for a forced job.)
	■ STAT – Static. Indicates that the condition (such as IMS-ACTIVE) is not date-dependent.
	Note: Before STAT was introduced, date 0101 was recommended to be used in conditions that were not date-dependent. Unlike 0101, STAT is not a date, and it operates differently. Always use STAT when defining conditions that are not date-dependent.
	**** - Any scheduling date. Valid only with opt set to - (Minus)
	■ \$\$\$\$ – Any scheduling date. Valid only with opt set to - (Minus)
	Note: If a date reference is not specified, the value ODAT is automatically inserted upon pressing <b>Enter</b> .

Table 173 DO COND Mandatory Subparameter Formats (part 2 of 2)

Subparameter	Description
opt	Indicates whether to add or delete the specified prerequisite condition. Valid values are:
	■ + (Plus) - Add (create) the prerequisite condition
	■ - (Minus) – Delete the prerequisite condition

### **General Information**

When a DO COND statement is activated, the specified prerequisite conditions are added to or deleted from the IOA Conditions file according to the value set for *opt*.

Prerequisite conditions are usually used to establish job dependencies or ensure manual intervention when required.

 To establish a job dependency, define a prerequisite condition in an OUT or DO COND statement in the job that must run first, and in an IN statement in the job that must run afterwards.

The job containing a prerequisite condition in its IN statement is not submitted unless that prerequisite condition has been added manually or by the job containing an OUT or DO COND statement.

An OUT statement is used to add the prerequisite condition if the job ends OK.

Use a DO COND statement to add the prerequisite condition if the step and code event criteria specified in the accompanying ON statement are satisfied.

■ If the IN prerequisite condition can only be satisfied by manual intervention (for example, TAPE1-ARRIVED is set by the operator after an external tape has arrived insight), performance of the required manual intervention before job submission is ensured.

OUT and DO COND statements can also be used to delete prerequisite conditions. The OUT statement of the job can be used to delete the prerequisite condition after the job ends OK. A DO COND statement can be used to delete prerequisite conditions if the accompanying ON step and code criteria are satisfied.

These statements are generally used to delete prerequisite conditions either to prevent a particular job from running or when the condition is no longer needed by any other jobs in the Active Jobs file.

DO COND functions are performed after the functions of the OUT parameter.

- If a prerequisite condition is added by the OUT parameter and deleted by the DO COND parameter, the combined effect is the deletion of the prerequisite condition.
- If a prerequisite condition is deleted by the OUT parameter and added by the DO COND parameter, the combined effect is the addition of that prerequisite condition.

The following are examples of prerequisite conditions:

IMS-ACTIVE
JOB_PAYCALC_ENDED_OK
TAPE1_LOADED

All prerequisite conditions are associated with a date reference that is used to distinguish between different runs of the same job with different scheduling dates. If, for example, a condition is being deleted, only the condition matching the specified date is deleted. The same condition associated with a different date is not deleted. When adding or deleting prerequisite conditions, the date associated with the prerequisite condition can be a specific 4-character date or one of the symbolic dates described in the explanation of parameter "dateref" on page 437.

Prerequisite conditions created by a DO COND statement can trigger the execution of other jobs or processes.

Prerequisite conditions deleted by a DO COND statement can prevent the execution of jobs and processes whose IN statements require those prerequisite conditions.

If two or more DO COND statements are contradictory, statements performed earlier are overridden by statements that are performed later.

For more information regarding prerequisite conditions, see "IN: Runtime Scheduling Parameter" on page 500, "ON Statements: Post–Processing Parameter" on page 537, and "OUT: Post–Processing Parameter" on page 562, and see "Prerequisite Conditions" on page 69.

### **Specifying Long DO COND Condition Names**

Regular prerequisite conditions are not more than 20 characters in length. If you want to specify a longer condition name, up to 39 characters in length, enter the string **LONG** in the date reference field of an empty DO COND condition line. An (L) appears at the beginning of the line. If the field already contains data, entering the string **LONG** will open a new long DO COND parameter, with (L) appearing at the beginning of the line. You can now insert a long condition name, as illustrated in Figure 178.

Specify **SHRT** in the date reference field to revert back to condition names of standard length.

#### NOTE



Long condition names cannot be used in CMEM rule definitions.

#### Figure 178 Long DO COND Condition

```
JOB: IEFBR14 LIB CTMP.V610.SCHEDULE
                                                                 TABLE: PHILL1
COMMAND ===>
                                                                 SCROLL===> CRSR
  IN CTM-DAILYPRD-ENDEDXX ODAT CTM-DAILYSYS-ENDEDXX ODAT
  CONTROL
  RESOURCE
  PIPE
  FROM TIME + DAYS UNTIL TIME + DAYS
DUE OUT TIME + DAYS PRIORITY 00 SAC CONFIRM
  TIME ZONE:
  OUT
  AUTO-ARCHIVE Y SYSDB Y MAXDAYS MAXRUNS
RETENTION: # OF DAYS TO KEEP # OF GENERATIONS TO KEEP
  SYSOUT OP (C,D,F,N,R)
MAXRERUN RERUNMEM
  MAXRERUN RERUNMEM INTERVAL
STEP RANGE FR (PGM.PROC) . TO
ON PGMST +EVERY PROCST STEP1 CODES SOC4
                                                      INTERVAL
                                                                     FROM
                                                                         A/0
    DO COND (L) THIS-IS-A-LONG-DO-COND-CONDITION-NAME ODAT PROCST CODES
  ON PGMST
                                                                        A/0
    D0
  SHOUT WHEN NOTOK TIME + DAYS TO OPER2
    MS LOADING OF MANUAL CONDITIONS SCREEN FAILED. CALL OP MANAGER.
  SHOUT WHEN TIME + DAYS TO
                                                                     URGN
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
                                                                        16.30.17
```

### Differences between OUT and DO COND

OUT and DO COND statements have the following differences:

- An OUT statement is applied only if the job ends OK. DO COND statements are associated with accompanying ON statements and are applied only if the accompanying ON step and code criteria are satisfied.
- An OUT statement appears in each job scheduling definition. No DO COND statement appears unless specified. To specify a DO COND statement, type COND in an empty DO field and press Enter.
- DO COND statements are processed after OUT statements and can therefore override OUT statements.
- MVS restart can only be requested from an OUT statement, not a DO COND statement.

## **Example**

The following example provides a simplified demonstration of how CONTROL-M can be used to monitor IMS. Prerequisite conditions, CHANGE-ACCUMULATION and LOGCLOSE-NEEDED, can be used as IN prerequisite conditions to trigger the execution of IMS maintenance jobs that depend on those conditions.

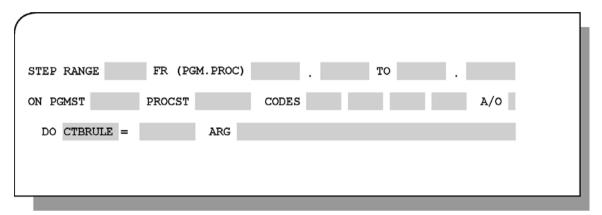
Figure 179 DO COND Parameter

JOB: IMSPROD LIB CTM.PROD.SCHEDULE	TABLE: IMSPROD
COMMAND ===>	SCROLL===> CRSR
+	+
OUT IMS-ACTIVE **** -	
AUTO-ARCHIVE Y SYSDB Y MAXDAYS MAXRUNS	
RETENTION: # OF DAYS TO KEEP 030 # OF GENERATIONS TO KEEP	
SYSOUT OP (C,D,F,N,R)	FROM
MAXRERUN RERUNMEM INTERVAL	FROM
STEP RANGE FR (PGM.PROC) . TO	•
ON PGMST STEP01 PROCST CODES U0421	A/0
DO COND CHANGE ACCUMULATION ODAT +	
DO	
ON PGMST STEP01 PROCST CODES U0428	A/0
DO COND LOGCLOSE-NEEDED ODAT +	
DO	A /O
ON PGMST STEPO1 PROCST CODES U0426	A/0
DO SHOUT TO U-DBA URGENCY V	
= *** IMSPROD ABENDED WITH U0426 ****	
	A / O
ON PGMST PROCST CODES	A/0
SHOUT WHEN TIME + DAYS TO	URGN
COMMANDS: EDIT, DOC, PLAN, JOBSTAT	11.17.00
COMMANDS. LDII, DOC, FLAN, OUDSTAT	11.17.00

# **DO CTBRULE: Post-Processing Parameter**

Invokes a CONTROL-M/Analyzer rule to be executed during the processing of a specific program step. Available only at sites utilizing CONTROL-M/Analyzer.

Figure 180 DO CTBRULE Parameter Format



Optional. Type CTBRULE in the DO field and press **Enter**. The DO CTBRULE subparameters are described in Table 174.

**Table 174 DO CTBRULE Subparameters** 

Subparameter	Description
rule_name	Name of the CONTROL-M/Analyzer rule that is to be executed. The CONTROL-M/Analyzer rule contains all balancing specifications to be performed. The rule name can be a maximum of eight characters. Mandatory.
arg	Arguments that are passed to the CONTROL-M/Analyzer rule. Separate multiple arguments by commas. A maximum of 45 characters can be specified. Optional.

### **General Information**

When DO CTBRULE is specified, balancing is performed by the CONTROL-M/Analyzer Runtime environment according to the specified rule definition and using the specified arguments. The CONTROL-M/Analyzer Runtime environment is invoked once for each DO CTBRULE statement in the job scheduling definition.

#### **NOTE**



If DO CTBRULE is specified under ON PGMST ANYSTEP, the CONTROL-M/Analyzer Runtime environment is invoked only once.

When CONTROL-M calls a CONTROL-M/Analyzer rule, the CONTROL-M/Analyzer System variable SYSOPT contains the value CTMWORK. This variable can then be tested within the CONTROL-M/Analyzer rule definition to determine if CONTROL-M invoked the CONTROL-M/Analyzer Runtime environment.

When the CONTROL-M/Analyzer Runtime environment is invoked by CONTROL-M, that is, the CONTROL-M/Analyzer System variable SYSOPT is set to CTMWORK, CONTROL-M/Analyzer can analyze and balance SYSDATA. For more information about invoking CONTROL-M/Analyzer rules from CONTROL-M job scheduling definitions, see the discussion of the interface to CONTROL-M in the *CONTROL-M/Analyzer FOR z/OS User Guide*.

#### **Example**

If the job ends OK, execute CONTROL-M/Analyzer balancing rule GOVTBAL.

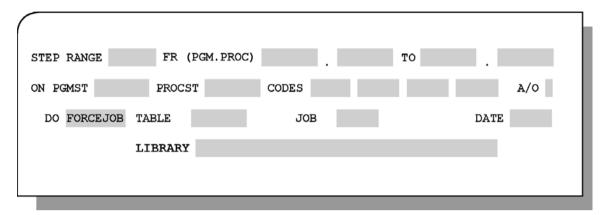
Figure 181 DO CTBRULE Parameter Example

MMAND ===>	IB CTM.PROD.SCF				TABLE: BACKUP SCROLL===> CRSR
	UNTIL				
AUTO-ARCHIVE	NCE-GOVTREPT-OH Y SYSI OF DAYS TO KE	В Ү			
SYSOUT OP ( MAXRERUN	C,D,F,N,R) RERUNMEM	,		INTERVAL	FROM FROM
ON PGMST ANYS	FR (PGM.F TEP PROCST = GOVTBAL AF	CODE	S OK		A/O
	PROCST	CODE	S		A/0
SHOUT WHEN NO	TOK TIME REPT ENDED "NO ⁻		DAYS TO	TS0-M44	URGN R
SHOUT WHEN MS	TIME	+	DAYS TO		URGN
>>>>>>	>>>>> E1	ID OF SCHED	ULING PARAM	IETERS <<<<<	<<<<<< ======
MMANDS: EDIT,	DOC, PLAN, JOE	STAT			11.17.00

# **DO FORCEJOB: Post-Processing Parameter**

Force one or more jobs under CONTROL-M.

Figure 182 DO FORCEJOB Parameter Format



Optional. Type FORCEJOB in the DO field and press **Enter**. The DO FORCEJOB subparameters are described in Table 175.

**Table 175 DO FORCEJOB Subparameter Formats** 

Subparameter	Description
TABLE	Name of CONTROL-M scheduling table.
JOB	Job name. If this field is blank, all jobs in the specified table are forced.
DATE	6-character scheduling date for the jobs. Valid values are:
	<ul> <li>date – Specific date (in either mmddyy, ddmmyy or yymmdd format, depending on the site standard).</li> </ul>
	■ ODAT – Inherit the original CONTROL-M scheduling date of the job that issues the DO FORCEJOB command. Default.
	■ DATE – Resolves to the current system date.
LIBRARY	Name of the scheduling library containing the specified table. The library name may contain auto-edit variables.

### **General Information**

The DO FORCEJOB statement schedules jobs under CONTROL-M even if the jobs are not normally scheduled on the specified date (according to the Basic Scheduling parameters of the job). It is similar to the FORCE option in the CONTROL-M Rule List screen or Table List screen.

If the DO FORCEJOB statement specifies a job name belonging to multiple jobs in the table, the first job in the table with that job name is forced.

Without the DO FORCEJOB statement, emergency jobs and jobs that run in special circumstances would require daily scheduling or manual forcing (from the Online facility). By defining appropriate ON criteria and DO FORCEJOB statements, emergency or other special jobs can be automatically forced when required without being previously scheduled.

The DO FORCEJOB statement causes the CONTROL-M monitor to dynamically allocate the job scheduling library specified in the LIBRARY parameter using the DD name ONSPLT.

#### **DO FORCEJOB request during RESTART**

The FORCONCE parameter in the CTMPARM member of the IOA PARM library may be set to control the behavior of DO FORCEJOB requests during a Restart of a job.

If DO FORCEJOB statements are not to be executed during the RESTART of a job if they were already executed during the original run of the job or during a previous RESTART of the job, then the FORCONCE parameter should be set to Y.

If DO FORCEJOB statements are always to be executed during the RESTART of a job (when the ON PGMST condition is satisfied), then the FORCONCE parameter should be set to N.

#### Failure of a DO FORCEJOB request

When a DO FORCEJOB request fails because the scheduling table is in use, CONTROL-M may try again to execute the job, depending on the values set for the FORCE#RT and FORCE#WI installation parameters. For more information on the FORCE#RT and FORCE#WI installation parameters, see the customization chapter of the INCONTROL for z/OS Installation Guide.

When a DO FORCEJOB request fails because the scheduling table has been migrated, the action of CONTROL-M depends on the value of the SCRECALL parameter in the CTMPARM member.

- If the value of SCRECALL is Y, the scheduling table is recalled, and CONTROL-M tries to reorder the job in the same way as if the scheduling table is in use.
- If the value of SCRECALL is N, the scheduling table is not recalled, and the job is not scheduled. This is the default setting of SCRECALL.

For more information on the SCRECALL parameter, see the customization chapter of the INCONTROL for z/OS Installation Guide.

### **Example**

On any system or user abend on any step in job PRDKPL01, force emergency job PRDKPLSP.

Figure 183 DO FORCEJOB Parameter Example

# **Confirmation panel**

If the DFJCONF profile variable is set to Y, and the JOB parameter in the DO FORCEJOB request is blank, a confirmation panel is displayed when exiting the Job Scheduling Definition screen. The confirmation panel is displayed only once for each DO FORCEJOB statement.

Figure 184 DO FORCEJOB Confirmation Panel

```
THIS JOB CONTAINS ONE OR MORE DO FORCEJOB STATEMENTS.

WHEN THE JOB IS ORDERED:

ARE YOU SURE YOU WANT TO FORCE THE WHOLE TABLE IN THE

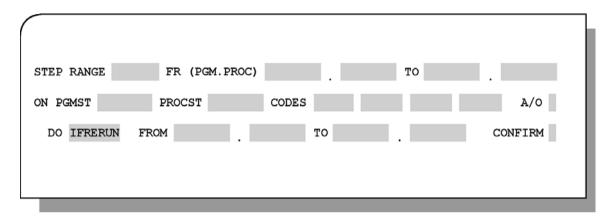
FORCEJOB STATEMENT(S)? (Y/N)
```

Enter Y to save the scheduling table and return to the Job List screen, or N to return to the scheduling table without saving it.

# §Restart§DO IFRERUN: Post–Processing Parameter

Job steps to be executed during restart of a job. Available only at sites utilizing CONTROL-M/Restart.

Figure 185 §Restart§ DO IFRERUN Parameter Format



Optional. Type IFRERUN in the DO field and press **Enter**. The DO IFRERUN subparameters are described in Table 176.

Table 176 §Restart§ DO IFRERUN Subparameter Formats (part 1 of 2)

Subparameter	Description
FROM	Step at which the job must be restarted. Mandatory. Valid values are:
	■ <i>pgmstep</i> – program step within the job stream (see the following note)
	■ <i>pgmstep.procstep</i> – program step within the called procedure (see the following note)
	■ SFIRST – first step of the job
	■ \$ABEND – step of the job that ended NOTOK due to system abend, user abend, condition code C2000 (PL/1 abend) or JFAIL (job failed on JCL error) \$ABEND is a subset of \$EXERR (below)
	■ \$FIRST.\$ABEND – first step of the abended procedure
	<ul> <li>\$FIRST.\$CLEANUP – this reserved keyword instructs CONTROL-M to run a CONTROL-M/Restart data set cleanup for the job</li> </ul>
	Data set cleanup is performed from the first step of the job. The job itself is not restarted.
	<ul> <li>\$EXERR – job step that ended with any error, including an abend, or that ended with a condition code that is redefined using the ON and DO statements as ENDED NOTOK</li> </ul>
	<b>Note</b> : If, during a restart of the job, the CONTROL-M/Restart step itself ends NOTOK, you must manually correct the problem encountered by the CONTROL-M/Restart step and then restart the job once again. In such a case, CONTROL-M/Restart does both of the following:
	<ul> <li>It restores DO IF RERUN decisions to their state before the last execution of the job.</li> <li>It deactivates the DO RERUN action.</li> </ul>
	This enables CONTROL-M/Restart to ignore the occurrence where the error arose during the operation of CONTROL-M/Restart itself. This prevents rerun processing from looping.

Table 176 §Restart§ DO IFRERUN Subparameter Formats (part 2 of 2)

Subparameter	Description
ТО	Step at which the restarted job must terminate. Optional. Valid values are:
	<ul> <li>pgmstep – Program step within the job stream (see the following note).</li> </ul>
	<ul> <li>pgmstep.procstep – Program step within the called procedure (see the following note).</li> </ul>
	If not specified, the restarted job terminates at the last job step that would normally be executed.
	Note: For both FROM and TO steps, <i>pgmstep</i> is the name of the step (EXEC statement) that executes the program from which to begin or end the restart:
	//pgmstep EXEC PGM=program
	<pre>procstep is the name of the step (EXEC statement) that invokes the procedure from which the above pgmstep program is executed:</pre>
	//procstep EXEC procedure
	pgmstep and procstep values can each be 1 through 8 characters.
	When specifying a <i>procstep</i> when the procedures are nested, the innermost <i>procstep</i> in which the program is included must be specified.
CONFIRM	Specifies whether a manual confirmation is required before the job is restarted. Valid values are:
	<ul> <li>Y (Yes) – Confirmation required. The job restart is not submitted unless a manual confirmation is entered in the Active Environment screen.</li> </ul>
	■ N (No) – No confirmation required. The job restart can be automatically submitted (by the DO RERUN statement) without a manual confirmation. Default.

#### **General Information**

When a DO IFRERUN statement is specified, the rerun is performed by the Restart facility of CONTROL-M/Restart using the specified restart subparameters.

- When DO IFRERUN is specified with a CONFIRM value of N (No):
  - If a DO RERUN statement follows, the job is automatically submitted for rerun.
  - If a DO RERUN statement does not follow, the job is not automatically rerun.
     Instead, the job remains displayed with its error status in the Active Environment screen.

In this case, to submit the job for rerun or restart, specify option R (Rerun) in the Active Environment screen. The Rerun (with Restart) Confirmation Window is displayed. Request the restart or rerun from the window.

■ When DO IFRERUN is specified with a CONFIRM value of Y (Yes), the job appears in the Active Environment screen with a WAIT CONFIRMATION (WITH RESTART) status and is not restarted unless confirmed. Specify option C (Confirm) to open the Confirm window to restart the job.

**§Restart§** For more information about job restart, see "Active Environment Screen" on page 167.

**§Restart§** When a job is submitted for restart, if \$FIRST is specified in the FROM subparameter, a \$FIRST step specification is passed "as is" to the CONTROLR step. If \$ABEND or \$EXERR is specified, the specified \$ABEND or \$EXERR value is first resolved to the appropriate step by the CONTROL-M monitor and then passed to the CONTROLR step. If \$FIRST.\$ABEND is specified, the CONTROL-M monitor determines which procedure abended and then passes the \$FIRST step specification for that procedure to the CONTROLR step. For information regarding the CONTROLR step, refer to the *CONTROL-M/Restart User Guide*.

**§Restart§** The CONTROL-M parameter MAXRERUN determines the maximum number of times the restart or rerun can be performed. For more information, see "MAXRERUN: Post–Processing Parameter" on page 516.

# **§Restart§ Example**

**§Restart§** If the job abends on any step, restart (and automatically rerun) the job from the first abended step.

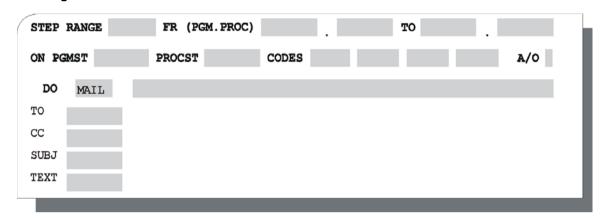
Figure 186 §Restart§ DO IFRERUN Parameter Example

IME: FROM	UNTIL	PRIORITY	DUE	OUT	SAC	CONFIRM
 )UT						
-	Y SYSI	)B Y	MAXDA	/S	MAXRUNS	
	OF DAYS TO KEE					
SYSOUT OP (	C,D,F,N,R)					FROM
1A X R E R U N	RERUNMEM			IN	TERVAL	FROM
	FR (PGM.				T0	
ON PGMST ANYS	TEP PROCST UPI	)A CODES	S****	U****	C2000	A/0
DO IFRERUN DO RERUN DO	FROM \$ABEND		T0			CONFIRM N
DN PGMST DO	PROCST	CODES				A/0
SHOUT WHEN MS	TIME	+ D	AYS	T0		URGN
>>>>>>	>>>>> E1	ND OF SCHEDU	LING PA	ARAMET	ERS <<<<<	<<<<<<< ===

# **DO MAIL: Post-Processing Parameter**

Send an e-mail message to the specified recipients.

Figure 187 DO MAIL Parameter Format



Optional. Type MAIL in the DO field and press Enter. The DO MAIL subparameters are described in Table 177.

Table 177 DO MAIL Subparameter Formats (part 1 of 2)

Subparameters	Description
ТО	<ul> <li>■ the full e-mail address</li> <li>■ the e-mail prefix</li> <li>If you use the prefix, the full e-mail address is supplied by the MAILDEST table in the IOA PARM library. The MAILDEST table also includes the value of the DFLTSFFX field, which specifies the e-mail address suffix (such as @MAIL.DOMAIN.COM), the SMTP STC name, and the HOST name.</li> <li>You can use AutoEdit variables in this field, in any combination of text and valid AutoEdit variables.</li> </ul>

Table 177 DO MAIL Subparameter Formats (part 2 of 2)

Subparameters	Description
CC	Destination to which a copy of the message is to be sent. Optional.  Valid values are:  ■ the full e-mail address ■ the e-mail prefix  If you use the prefix, the full e-mail address is supplied by the MAILDEST table in the IOA PARM library. The MAILDEST table also includes the value of the DFLTSFFX field, which specifies the e-mail address suffix (such as @MAIL.DOMAIN.COM), the SMTP STC name, and the HOST name.  You can use AutoEdit variables in this field, in any combination of text and valid AutoEdit variables.
SUBJ	Message subject of up to 70 characters. Optional.
TEXT	Message text of up to 255 text lines, each with a maximum of 70 characters. Optional.

#### **General Information**

The specified e-mail is sent to the specified destinations when the accompanying ON statement criteria are satisfied. Although e-mail can be sent using a DO SHOUT statement, the DO MAIL statement provides the following advantages:

- Using DO MAIL, you can specify any number of TO and CC recipients. With DO SHOUT, you must specify the mail destination prefix, and you must define the address in the MAILDEST table.
- Using DO MAIL, the e-mail text can exceed 70 characters. System variables and user-defined AutoEdit variables (Global IOA variables and variables defined in the SET VAR job or group scheduling definition parameter only) are supported in the subject line and message text. These variables are resolved when the DO MAIL statement is processed.

#### NOTE



The resolved value of an AutoEdit variable is truncated after 70 characters.

For information on the use of AutoEdit variables, see Chapter 5, "JCL and AutoEdit Facility."

If installation parameter ATTSYSOT=Y, the job's SYSOUT will be attached to the e-mail message.

#### **Subparameters TO and CC**

Each of these parameters can contain more than one mail name address. When a value is specified in the TO or CC field, a new empty line is displayed so that an additional value can be specified (up to a maximum of 255 lines).

Multiple addresses, separated by a semicolon (;), can be specified on a line.

If an address exceeds the length of a full line, it can be continued on the following line.

### **Example**

If the job is not run due to a JCL error, send an e-mail to the relevant users:

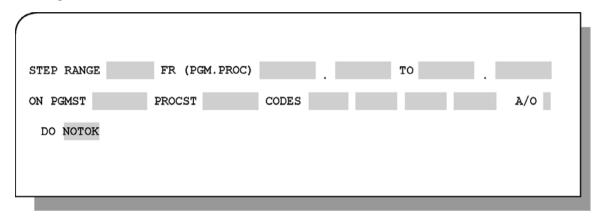
Figure 188 DO MAIL Parameter Example

```
JOB: SACALCO1 LIB CTM.PROD.SCHEDULE
                                                            TABLE: SALES
COMMAND ===>
                                                           SCROLL==> CRSR
  AUTO-ARCHIVE Y
                       SYSDB Y
                                     MAXDAYS
  RETENTION: # OF DAYS TO KEEP 030 # OF GENERATIONS TO KEEP
  SYSOUT OP (C.D.F.N.R)
                                                                  FROM
  MAXRERUN
            RERUNMEM
                                                 INTERVAL
                                                               FROM
  STEP RANGE
                   FR (PGM.PROC)
                                                  T0
  ON PGMST ANYSTEP PROCST CODES JNRUN
                                                                   A/0
   DO MAIL
  T0
      MAIL_ADDRESS_#1
      MAIL_ADDRESS_#2
     MAIL_ADDRESS_#3; MAIL_ADDRESS_#4
  SUBJ WARNING MESSAGE
  TEXT JCL ERROR IN SALES JOB! PLEASE CORRECT ERRORS AND RERUN THE JOB
   DO
  ON PGMST
                  PROCST
                                 CODES
                                                                   A/0
   D0
  SHOUT WHEN
                     TIME
                                     DAYS
                                                                URGN
  ---- >>>>>>>>>>> END OF SCHEDULING PARAMETERS <<<<<<<<<<<
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
```

# **DO NOTOK: Post-Processing Parameter**

Set the status of the job step to NOTOK if the accompanying ON step and code criteria are satisfied. If specified in a Group Entity, the termination status of the group is set to NOTOK.

Figure 189 DO NOTOK Parameter Format



Optional. Type NOTOK in the DO field and press **Enter**. DO NOTOK has no subparameters.

#### **General Information**

A DO NOTOK statement can change the status of a job step from OK to NOTOK. This results in the job having a final termination status of ENDED NOTOK.

When specified in a Group Entity, a DO NOTOK statement changes the termination status of the group (not the status of jobs or job steps).

The following paragraphs describe the relationship of job step status and the final termination status of the job.

- CONTROL-M determines the status of each individual step in a job before determining the final status of the job.
- After examining the results of a job step, CONTROL-M automatically assigns a status of OK or NOTOK to the step:
  - By default, any job step ending with a condition code of C0000 through C0004 is assigned a status of OK, but the INCONTROL administrator can change this.
  - If any other condition code, system or user abend code, or user event is generated, the step is automatically assigned a status of NOTOK.

■ In general, if any of the steps in a job ends with a status of NOTOK, the job is assigned a final status of ENDED NOTOK. For a job to be assigned a final status of ENDED OK, each step in the job must be assigned a status of OK.

This logic suits most situations. Do not change it. However, there may be a situation in which CONTROL-M assigned a step a status of OK, but the status ought to be changed to NOTOK. Such a situation is described in the following example. The job ended with a condition code of C0004, but in this particular situation, it is better that the step have a status of NOTOK and the entire job be assigned a status of ENDED NOTOK.

DO NOTOK cannot be specified for the same ON step and code event as DO OK.

When a DO NOTOK statement is performed for a step, the final status of the job is ENDED NOTOK, even if was previously set to ENDED OK.

#### _ NOTE



I

A DO NOTOK statement is ignored if it is specified in an ON PGMST +EVERY statement.

### **Example**

When PROCST UPDA in PGMST STEP06 finishes executing with a condition code of C0004, it is considered NOTOK.

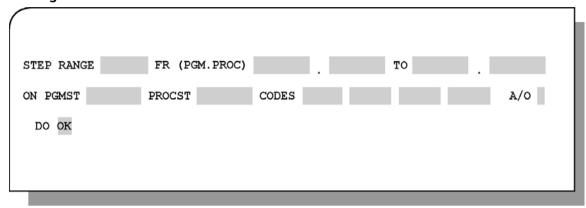
Figure 190 DO NOTOK Parameter Example

```
JOB: PRDKPLO2 LIB CTM.PROD.SCHEDULE
                                                                 TABLE: PRODKPL
COMMAND ===>
                                                                 SCROLL===> CRSR
  OUT
  AUTO-ARCHIVE Y
                          SYSDB Y
                                          MAXDAYS
  RETENTION: # OF DAYS TO KEEP 030 # OF GENERATIONS TO KEEP
  SYSOUT OP (C,D,F,N,R)
MAXRERUN RERUNMEM
STEP RANGE FR (
                                                                        FROM
                                                      INTERVAL
                                                                     FROM
                    FR (PGM.PROC)
                                                        T0
  ON PGMST STEP06 PROCST UPDA
                                    CODES COOO4
                                                                        A/0
    DO NOTOK
    D0
  ON PGMST
                    PROCST
                                     CODES
                                                                         A/0
    D0
  SHOUT WHEN
                       TIME
                                         DAYS
                                                  T0
                                                                      URGN
 ----- >>>>>>>>>> END OF SCHEDULING PARAMETERS <<<<<<<<<< ----
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
                                                                        15.16.03
```

# **DO OK: Post–Processing Parameter**

Set the status of the job step to OK if the accompanying ON step and code criteria are satisfied. If specified in a Group Entity, the termination status of the group is set to OK.

Figure 191 DO OK Parameter Format



Optional. Type OK in the DO field and press Enter. DO OK has no subparameters.

#### **General Information**

A DO OK statement can change the status of a job step from NOTOK to OK. If all steps of a job have a status of OK the job is assigned a final termination status of ENDED OK.

When specified in a Group Entity, a DO OK statement changes the termination status of the group (not the status of jobs or job steps).

The relationship between job step status and the final termination status of the job is as follows:

CONTROL-M determines the status of each individual step in a job before determining the final status of the job.

After examining the results of a job step, CONTROL-M automatically assigns a status of OK or NOTOK to the step:

- By default, any job step ending with a condition code of C0000 through C0004 is assigned a status of OK, but the INCONTROL administrator can change this.
- If any other condition code, system or user abend code, or user event is generated, the step is automatically assigned a status of NOTOK.

In general, if any of the steps in a job ends with a status of NOTOK, the job is assigned a final status of ENDED NOTOK. For a job to be assigned a final status of ENDED OK, each step in the job must be assigned a status of OK.

This logic suits most situations. Do not change it. However, there may be a situation in which CONTROL-M assigned a step a status of NOTOK, but the status ought to be changed to OK. Several such exceptional situations are described below:

- The NOTOK status is inappropriate for the job step. For example, a condition code greater than C0004 was returned for a step that had an acceptable result.
- The NOTOK status is appropriate for the job step, but the job step is not critical, and ought not to affect the final job status.
- User events created using Exit CTMX003 always result in a NOTOK status unless DO OK is specified.

DO OK cannot be specified for the same ON step and code event as DO NOTOK and DO RERUN.

A DO OK statement specified in the job scheduling definitions is ignored if one of the following occurs:

- any of the following status codes apply to any step:
  - EXERR
  - -JNSUB
  - -*REC0
  - -*UKNW
- it was specified as part of an ON PGMSTEP ANYSTEP ..... CODE NOTOK condition, because if that condition is satisfied, the status of the job has already been set to NOTOK
- it is specified in an ON PGMST +EVERY statement

For more information, see "Valid CODES Values" on page 551.

# **Example**

When PROCST UPDA in PGMST STEP08 finishes executing with a condition code less than C0008, it is considered OK.

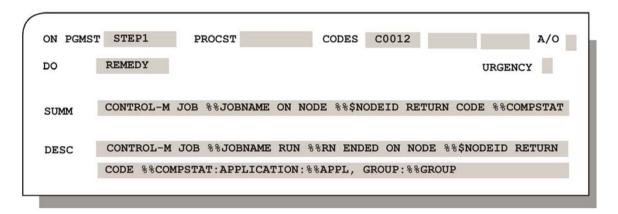
Figure 192 DO OK Parameter Example

JOB: PRDKPLO2 LIB CTM.PROD.SCHEDULE COMMAND ===>	TABLE: PRODKPL SCROLL===> CRSR
OUT AUTO-ARCHIVE Y SYSDB Y MAXDAYS MAXRUNS RETENTION: # OF DAYS TO KEEP 030 # OF GENERATIONS TO KEEP SYSOUT OP (C,D,F,N,R) MAXRERUN RERUNMEM INTERVAL STEP RANGE FR (PGM.PROC) . TO ON PGMST STEP08 PROCST UPDA CODES <c0008< th=""><th>FROM</th></c0008<>	FROM
DO OK DO ON PGMST PROCST CODES DO	A/0
SHOUT WHEN TIME + DAYS TO  MS  >>>>>>>> END OF SCHEDULING PARAMETERS <<<<<	URGN
COMMANDS: EDIT, DOC, PLAN, JOBSTAT	15.16.03

# **DO REMEDY: Post-Processing Parameter**

Open a problem ticket within the Remedy Helpdesk System, for example, when a specific job fails.

Figure 193 DO REMEDY Parameter Format



Optional. Type **REMEDY** in the DO field and press **Enter**. The DO REMEDY subparameters are described in Table 178.

**Table 178 DO REMEDY Subparameter Formats** 

Subparameters	Description
URGENCY	Urgency of the message. Mandatory. Valid values are:  ■ ' '(blank) – This is equivalent to L–Low. Default. ■ L–Low ■ M–Medium ■ H–High ■ U–Urgent
SUMM	Text, summarizing the description of the reason for opening a Remedy Helpdesk ticket, of up to 2 text lines, with a maximum 122 characters. Mandatory.  You can use any combination of text and valid AutoEdit variables.
DESC	Text, describing the reason for opening a Remedy Helpdesk ticket, of up to 15 text lines, with a maximum of 1018 characters. Mandatory.  You can use any combination of text and valid AutoEdit variables.

### **General Information**

An e-mail message is sent to the Remedy Helpdesk when the accompanying ON statement criteria are satisfied. To close the ticket, you must access the Remedy online service. You can use AutoEdit variables in the SUMM and DESC fields, in any combination of text and valid AutoEdit variables.

#### - NOTE -



The resolved value of an AutoEdit variable is truncated after 70 characters. For information on the use of AutoEdit variables, see Chapter 5, "JCL and AutoEdit Facility."

### **Example**

A job did not run due to a JCL error. This causes a Remedy Helpdesk ticket to be opened.

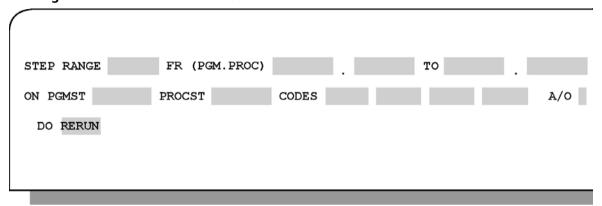
#### Figure 194 DO REMEDY Example

JOB: SACALCO1 LIB CTMP.V620.SCHEDULE  COMMAND ===> +	TABLE: SALES SCROLL===> CRSR
AUTO-ARCHIVE Y SYSDB Y MAXDAYS MAXRUNS	
RETENTION: # OF DAYS TO KEEP # OF GENERATIONS TO KEEP	FROM
SYSOUT OP (C,D,F,N,R) MAXRERUN RERUNMEM INTERVAL	
STEP RANGE FR (PGM.PROC) . TO   ON PGMST ANYSTEP PROCST CODES JNRUN   DO REMEDY URGENCY U	.   A/0
SUMM JCL ERROR IN SALES JOB	
DESC A JCL ERROR OCCURED IN THE SALES JOB. PLEASE CORRECT AN JOB.	D RERUN THE
   D0	
ON PGMST PROCST CODES	A/0
	0 132 A/0
SHOUT WHEN TIME + DAYS TO	URGN
COMMANDS: EDIT, DOC, PLAN, JOBSTAT	17.44.21

# **DO RERUN: Post-Processing Parameter**

Reschedules the job (for rerun) if the accompanying ON step and code criteria are satisfied.

Figure 195 DO RERUN Parameter Format



Optional. Type RERUN in the DO field and press **Enter**. DO RERUN has no subparameters.

#### **General Information**

The RERUN statement is intended for situations in which a job must be rescheduled following an unsuccessful job run.

Rerun jobs remain in the Active Jobs file with a status of WAIT SCHEDULE, where they wait to be submitted like any other job. However, other parameters, such as CONFIRM and DO IFRERUN, can affect the status of the rerun job order and the submission and processing of the job:

- A Y value specified in the CONFIRM parameter indicates that manual confirmation is required before submitting the rerun job order. In this case, the rerun job order is placed in the Active Jobs file with a status of WAIT CONFIRMATION (FOR SCHEDULE). The job can be confirmed by option C (Confirm) in the Active Environment screen.
- A DO IFRERUN statement before the DO RERUN statement indicates that a restart is desired instead of a full rerun. The job order is placed in the Active Jobs file with a status of WAIT SCHEDULE WITH RESTART, where the job waits to be submitted from the indicated restart step. Confirmation can also be required for restart jobs. This, too, is performed from the Active Environment screen. For more information, see "§Restart§DO IFRERUN: Post–Processing Parameter" on page 447.

For information about confirmation, see "Confirm Rerun Window" on page 222 and "\$Restart\$Rerun and/or Restart Window (Under CONTROL-M/Restart)" on page 224.

Job rerun is also affected by the MAXRERUN, RERUNMEM and INTERVAL parameters.

- MAXRERUN specifies the maximum number of times the job must be scheduled for rerun.
- RERUNMEM specifies the JCL member to be used for the rerun (if different from the normal JCL member of the job).
- INTERVAL specifies the number of minutes to wait between reruns.

These parameters are described in this chapter.

DO RERUN cannot be specified for a cyclic job or a cyclic started task.

DO RERUN cannot be specified for the same ON step and code event as DO OK.

Do not specify DO RERUN for steps that have a specified ON statement code value of OK.

Do not specify DO RERUN for steps that have a specified ON statement code value of NOTOK because many of the causes of a NOTOK status, such as JCL not found, preclude the possibility of a successful job rerun. Instead, specify an ON statement code value of EXERR to accompany the DO RERUN statement.

When a DO RERUN statement is performed for a job (that is, the accompanying ON step and code criteria are satisfied), the previously run job is automatically assigned a final status of ENDED NOTOK, even if the job would have otherwise had a status of ENDED OK.

# **Example**

If job EF145TS abends during step name COLLECT, try to run another job from member EF145TSR that continues from the same place.

Figure 196 DO RERUN Parameter Example

JOB: EF145TS LIB CTM. COMMAND ===>				TABLE: EFPROD SCROLL===> CRSR
0UT				
AUTO-ARCHIVE Y	SYSDB	Y MAXDAY	S MAXRUNS	
RETENTION: # OF DAY		# OF GENERA	TIONS TO KEEP	
SYSOUT OP (C,D,F,N				FROM
MAXRERUN RERUNM			INTERVAL	FROM
STEP RANGE F			T0	
ON PGMST COLLECT PF	(0021	CODE2 2***	0 * * * *	A/0
DO KEKUN				
= *	ROCST	CODES		A/0
DO				
SHOUT WHEN	TIME +	DAYS	T0	URGN
MS				
====== >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>	>>>> END OF S	CHEDULING PAR	AMETERS <<<<<	<<<<<< =====
COMMANDS: EDIT, DOC,	PLAN, JOBSTAT			11.17.00

# **DO SET: Post–Processing Parameter**

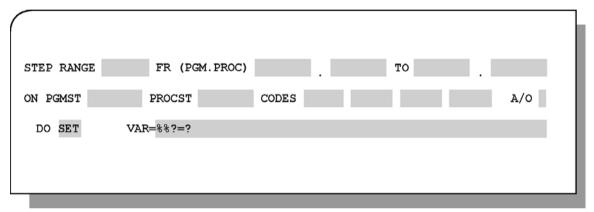
Assigns a value to an AutoEdit variable that can be used to set up the JCL of the next submission of a cyclic or rerun or restarted job, or to define a Global variable in the IOA Global Variable Database.

#### - NOTE -



DO SET and SET VAR statements are similar, but not identical. The differences are outlined in "Differences between DO SET and SET VAR" on page 467.

Figure 197 DO SET Parameter Format



Optional. Type SET in the DO field and press **Enter**. The DO SET subparameter is described in Table 179.

**Table 179 DO SET Subparameter** 

Subparameter	Description
VAR	User-defined variable and the value to be assigned. Mandatory.
	Replace the %%?=? prompt with the desired parameter, in the format:
	%%variable=expression
	For more information, see "SET VAR: General Job Parameter" on page 621.

#### **General Information**

A major advantage of using AutoEdit variables is that the JCL can be submitted with different values for different executions without actually changing the JCL.

There are two types of AutoEdit variables:

- System variables, which are assigned values by the system
- User-defined variables, for which the user must supply values; these variables can be either local or global

One method of supplying a value for a user-defined variable is by defining the variable and its value in a DO SET statement.

During job processing, the value is assigned at time of job submission. However, DO SET is a post-processing statement, which means that before it can be applied, its accompanying ON criteria must be satisfied in a job run.

Therefore, the DO SET statement is generally useful for supplying local user-defined variables for cyclic, rerun, or restarted jobs.

When the ON criteria of a DO SET statement that defines a local variable are satisfied during a job run, CONTROL-M creates a SET VAR statement equivalent to the DO SET statement (that is, containing the same variable and value) in the subsequent job run.

At time of job submission, AutoEdit variables in the JCL are resolved in the order in which they appear in the JCL. By default, if an AutoEdit variable cannot be resolved, the job is not submitted. This default can be changed using an appropriate %%RESOLVE AutoEdit control statement.

#### NOTE -



If the JCL contains an AutoEdit variable that is resolved in a subsequent run by a DO SET statement, the variable must be resolved by some other method, such as a SET VAR statement, in the original run, or the job is not submitted.

DO SET statements can also be used to define and update Global Variables in the IOA Global Variable database. The database is updated as part of job post-processing, when the DO SET statement is processed. For more information on Global Variables, including Global Variable syntax, see Chapter 5, "JCL and AutoEdit Facility."

An unlimited number of DO SET statements can be specified.

JCL Setup and the AutoEdit facility are described in depth in Chapter 5, "JCL and AutoEdit Facility."

### **Differences between DO SET and SET VAR**

DO SET and SET VAR statements are similar but have the following differences:

- Local variables in SET VAR statements are always applied before the job is submitted. DO SET is a post-processing statement that can only be applied after its accompanying ON step and code criteria are satisfied. This means that a local value specified in the DO SET statement can only be applied in the next submission of the job (specifically, for cyclic and rerun or restarted jobs).
- Global variables specified in a SET VAR statement are defined or updated in the IOA Global Variable database before job submission. Global variables specified in a DO SET statement are defined or updated in the IOA Global Variable database as part of job post-processing
- The SET VAR statement appears in each job scheduling definition. The DO SET statement does not appear unless specified. To specify a DO SET statement, type SET in an empty DO field and press **Enter**.
- In the SET VAR statement, the parameter value is specified after the keyword VAR. In the DO SET statement, the parameter value is specified after the keyword VAR=.

# **Example**

If the job execution fails on any step due to a system or user abend, resolve the %%PARM parameter in the JCL to RESTART, restart from the first abended step, and automatically rerun.

Figure 198 DO SET Parameter Example

JOB: PRDKPL01 LIB CTM.PROD.SCHEDULE	TABLE: PRODKPL
COMMAND ===>	SCROLL===> CRSR
OUT PRDKPLO1-ENDED-OK ODAT +	
AUTO-ARCHIVE Y SYSDB Y MAXDAYS MAXRU	
RETENTION: # OF DAYS TO KEEP 030 # OF GENERATIONS TO KEE	
SYSOUT OP (C,D,F,N,R)	FROM
	L FROM
STEP RANGE FR (PGM.PROC) . TO	
ON PGMST ANYSTEP PROCST CODES S*** U**** C200	
DO IFRERUN FROM \$ABEND . TO . DO RFRUN	CONFIRM N
DO SET VAR= %%PARM=RESTART	
DO DO	
ON PGMST PROCST CODES	A/0
DO	
SHOUT WHEN TIME + DAYS TO	URGN
MS	
====== >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>	<<<<<<< =====
COMMANDS: EDIT, DOC, PLAN, JOBSTAT	11.17.00

## **DO SHOUT: Post-Processing Parameter**

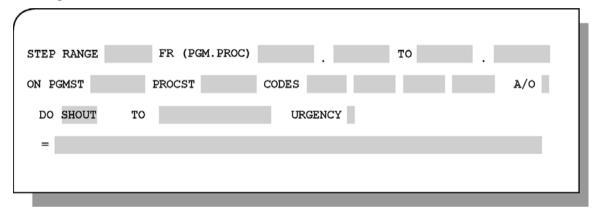
Specifies a message to be sent ("shouted") to a specific destination when a specific situation occurs.



#### NOTE

DO SHOUT and SHOUT statements are similar, but not identical. The differences are outlined in "Differences between SHOUT and DO SHOUT" on page 474.

Figure 199 DO SHOUT Parameter Format



Optional. Type SHOUT in the DO field and press **Enter**. The DO SHOUT subparameters are described in Table 180.

Table 180 DO SHOUT Subparameters (part 1 of 2)

Subparameter	Description
ТО	Destination of the message (1 through 16 characters). Mandatory. Valid values are:
	■ U-userid or USERID-userid – Writes the message to the IOA Log file under the specified user ID. userid must be 1 through 8 characters.
	■ OPER[-n] – Sends a rollable message to the operator console. n is an optional 2-digit route code. If a route code is not specified, the default routes are Master Console and Programmer Information (1 and 11), and optionally, CONTROL-M/Enterprise Manager. For more detailed information regarding route codes, refer to the IBM publication <i>Routing and Descriptor Codes, GC38-1102</i> .
	■ OPER2[-n] – Sends an unrollable, highlighted message to the operator console. n is an optional 2-digit route code. If a route code is not specified, the default routes are Master Console and Programmer Information (1 and 11), and optionally, CONTROL-M/Enterprise Manager. For more detailed information regarding route codes, refer to the IBM publication <i>Routing and Descriptor Codes, GC38-1102</i> .
	■ [TSO - loginid   T - loginid] [;Nn   ;Mm   ;NnMm   ;Lname] – Sends the message to the specified ID (groupid or logonid). ID is mandatory.  If a groupid is specified, it must be a valid ID found within the IOA Dynamic Destination Table.  If a logonid is specified, it must be 1 through 7 characters.  An optional second value, indicating the computer and/or node (such as Mm) of the TSO logonid, can be specified, as follows:  Under JES2:  Valid values are: Nn, Mm or NnMm, where:  - m is the machine ID (the computer in JES2, not the 4-character SMF system ID). For more information, see the discussion on specifying IOA CPUs in the description of the customization process in the INCONTROL for z/OS Installation Guide.  - n is the 1 to 2 character JES/NJE node ID.  Under JES3:
	The only valid value is <i>Lname</i> , where <i>Lname</i> is the logical JES name of the machine (that is, the name as used in JES3 command *T, not the SMF system ID).  For more information, see the discussion on specifying IOA CPUs in the description of the customization process in the <i>INCONTROL for z/OS Installation Guide</i> .
	<b>Note</b> : A shout to a TSO user performs a TSO SEND command, which may require authorization at the receiving node.

Table 180 DO SHOUT Subparameters (part 2 of 2)

Subparameter	Description
	■ U-M: <i>mail-name-prefix</i> – Sends a message by mail to the recipient identified by <i>mail-name-prefix</i> (1 through 12 characters).
	<ul> <li>U-S:snmp_dest - Sends an SNMP trap (message) to the recipient identified by snmp_dest.</li> <li>snmp_dest consists of from 1 through 12 characters, and can be any of the following:         <ul> <li>a host name</li> <li>an IP address</li> <li>a nickname defined in the SNMPDEST destination table</li> <li>a group name defined in the SNMPDEST destination table</li> <li>U-ECS - Sends messages to the CONTROL-M/Enterprise</li> </ul> </li> </ul>
	Manager user. For more information on this feature, see "Shouting to CONTROL-M/Enterprise Manager" on page 473.
URGENCY	Determines the priority level of the message. Valid values are:  ■ R − Regular. Default. ■ U − Urgent ■ V − Very urgent
=	Message text Maximum length: 70 characters AutoEdit variables (both system and user-defined) are supported and automatically resolved (replaced) at the time the SHOUT message is issued. For AutoEdit usage information, see Chapter 5, "JCL and AutoEdit Facility."

#### **General Information**

The message is sent to the required destination when the accompanying ON statement criteria are satisfied.

DO SHOUT statements can also be defined in Group entities, where they are used in a manner similar to jobs.

#### The TO Subparameter

Specify TO=USERID-userid to write the message to the IOA Log file under the user ID specified in the parameter.

Specify TO=OPER[-n] to send the message to the operator console (route code n). If the n value is omitted, the message is sent to all consoles to which route codes 1 or 11 are assigned. For more detailed information regarding route codes, refer to the IBM publication Routing and Descriptor Codes, GC38-1102. Optionally, the message can also be sent to the CONTROL-M/Enterprise Manager user, as described in "Shouting to CONTROL-M/Enterprise Manager" on page 473.

Specify TO=OPER2[-n] to send a highlighted, unrollable message to the operator console (route code n). If the n value is omitted, the message is sent to all consoles to which route codes 1 or 11 are assigned. For more detailed information regarding route codes, refer to the IBM publication Routing and Descriptor Codes, GC38-1102. Optionally, the message can also be sent to the CONTROL-M/Enterprise Manager user, as described in "Shouting to CONTROL-M/Enterprise Manager" on page 473.

Specify TO=TSO-id or T-id to send the message to a groupid or logonid. The Shout facility first searches the IOA Dynamic Destination table for the specified ID. If the table contains an entry (groupid) that matches the value, the content of the entry is used as the target for the shouted message. The entire TO field is used. Therefore, when directing the message to a remote user, do not append Nn or Mm. Instead, do this in the IOA Dynamic Destination Table itself. For more information, see the description of the Dynamic Destination Table in the *INCONTROL for z/OS Administrator Guide*.

If no matching ID is found in the Dynamic Destination table, the Shout facility assumes the specified ID is a logonid. It then creates a TSO message that it hands over to MVS. MVS then sends the message to that logonid. (If the logonid does not exist, MVS cannot send the message, but no error message is generated.) When a second value is used, the message is sent to the TSO logonid in the specified computer or node (machine ID). To determine the machine ID under JES2, specify JES command SD MEMBER.

Specify TO=U-M: mail-name-prefix to send the message by e-mail to the recipient identified by the prefix. The full mail name address is supplied by the MAILDEST table in the IOA PARM library. For more information about mail destinations, see the INCONTROL for z/OS Administrator Guide. The MAILDEST table also includes DFLTSFFX, the mail address suffix, such as @MAIL.DOMAIN.COM, the SMTP STC name, and the HOSTNAME. If installation parameter ATTSYSOT=Y, the job's SYSOUT will be attached to the e-mail message.

#### NOTE -



When you update the MAILDEST table in the IOA PARM library, ensure that you distinguish between lower case and upper case characters.

Specify TO=U-S:snmp_dest to send the SNMP trap (message) to the recipient identified by snmp_dest. This variable (snmp_dest) can be any of the following:

- a host name
- an IP address
- a nickname defined in the SNMPDEST table
- a group name defined in the SNMPDEST table

For more information about mail destinations, see the *INCONTROL* for z/OS *Administrator Guide*.

#### Shouting to CONTROL-M/Enterprise Manager

For CONTROL-M to be able to shout to CONTROL-M/Enterprise Manager, the following conditions must be satisfied at the site:

- 1. CONTROL-M/Enterprise Manager must be installed and the ECS parameter must be set to Y in member IOAPARM in the IOA PARM library.
- 2. File MG2 (the CONTROL-M/Enterprise Manager Shout File) must be defined.
- 3. The following parameters in the IOAPARM member in the IOA PARM library must be defined according to how messages are targeted to CONTROL-M/Enterprise Manager:
  - If TO=OPER and TO=OPER2 messages must be sent to CONTROL-M/Enterprise Manager, the OPER2ECS parameter must be set to Y (Yes). Otherwise, it must be set to N (No).

#### When OPER2ECS is set to Y:

- If these messages must also be sent to the MVS operator console, the OPER2CON parameter must also be set to Y (Yes).
- If these messages must not also be sent to the MVS operator console, the OPER2CON parameter must also be set to N (No).
- If TO=U-ECS messages must be sent to CONTROL-M/Enterprise Manager, the UECS2ECS parameter must be set to Y (Yes); otherwise, it must be set to N (No). Regardless of the value of this parameter, these messages are also sent to CONTROL-M and the IOA Log.

Once the above conditions are satisfied, messages can be shouted to CONTROL-M/Enterprise Manager by specifying a destination of TO=OPER or TO=OPER2 (without a route code qualifier), or TO=U-ECS.

Such messages are then placed by CONTROL-M in the M2G file. Once the shouted message is in the M2G file, the CONTROL-M Application Server reads the file and sends the message to the CONTROL-M/Enterprise Manager user.

#### The URGENCY Subparameter

The URGENCY value indicates the urgency level of the message.

In addition, if the destination is USERID-*userid* (or U-*userid*), the user can control, according to urgency, which messages are displayed when the IOA Log file is accessed. Urgent and very urgent messages are highlighted on the screen. For more details, see "IOA Log Facility" on page 296

### **Differences between SHOUT and DO SHOUT**

SHOUT and DO SHOUT statements have the following differences:

■ A DO SHOUT statement is applied only if the accompanying ON criteria are satisfied. Therefore a DO SHOUT statement does not contain subparameters for specifying when to perform the shout.

By contrast, a SHOUT statement requires that a value be specified in subparameter WHEN indicating when to shout the message. Messages can be shouted when the job ends OK or NOTOK, when the job is late for submission or completion, or when the job runs too long.

- A SHOUT statement appears in each job scheduling definition. A DO SHOUT statement does not appear unless specified. To specify a DO SHOUT statement, type SHOUT in an empty DO field and press Enter.
- The SHOUT URGN subparameter is equivalent to the DO SHOUT URGENCY subparameter.
- The SHOUT MS subparameter is equivalent to the DO SHOUT subparameter.

## **Example**

If the job is not run because of a JCL error, notify the user who sent the job.

Figure 200 DO SHOUT Subparameter Example

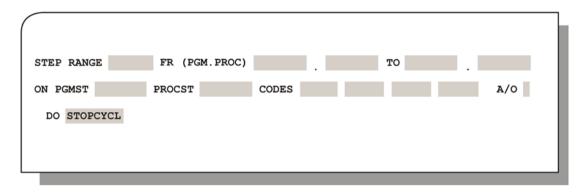
```
JOB: SACALCO1 LIB CTM.PROD.SCHEDULE
                                                              TABLE: SALARY
COMMAND ===>
                                                              SCROLL===> CRSR
  OUT
                        SYSDB Y MAXDAYS
  AUTO-ARCHIVE Y
  RETENTION: # OF DAYS TO KEEP 030 # OF GENERATIONS TO KEEP
 SYSOUT OP (C,D,F,N,R)
MAXRERUN RERUNMEM
STEP RANGE FR (P
                                                                    FROM
                                                 INTERVAL
                                                                 FROM
                   FR (PGM.PROC)
  ON PGMST ANYSTEP PROCST CODES JNRUN
DO SHOUT TO TSO-U0014 URGENCY U
                                                                     A/0
    = *** JCL ERROR IN SALARY JOB ***
  ON PGMST PROCST
                                                                     A/0
                                  CODES
   D0
                                                                   URGN
  SHOUT WHEN
                    TIME + DAYS
   MS
   ==== >>>>>>>>> END OF SCHEDULING PARAMETERS <>>>>>>>>>
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
                                                                    11.17.00
```

An urgent message is sent to the user ID that requested the job.

## **DO STOPCYCL: Post-Processing Parameter**

Stops recycling a cyclic task.

Figure 201 DO STOPCYCL Parameter Format



The DO STOPCYCL statement has no subparameters. Type the word STOPCYCL in the DO field and press **Enter**.

#### **General Information**

DO STOPCYCL is intended for use with all cyclic tasks (cyclic job, cyclic STC, emergency cyclic job and emergency cyclic STC). It interrupts a job cycle after the current run, so that once the job completes its current run, it does not run again. This parameter does not change the status (OK or NOTOK) assigned to job during the last cycle.

After DO STOPCYCL has interrupted a job, commands RERUN or RESTART can be used in the Active Environment screen to continue the job cycle from where it stopped. Commands RERUN and RESTART resume the stopped cyclic tasks without waiting for a cycling interval to occur. After the job restarts, it continues its normal cyclic interval as before.

## **Example**

If cyclic job SACALCO1 finishes with a status of NOTOK, the STOPCYCL parameter interrupts the cycle.

Figure 202 DO STOPCYCL Parameter Example

UT						
	SYSDE					
	F DAYS TO KEEP	P 030 #	OF GENE	RATIONS	TO KEEP	
YSOUT OP (C,					THE DULL OO	FROM
AXRERUN RE		2001			INTERVAL 003	FRUM
	FR (PGM.PF P PROCST		DEC NOTO		T0	A/O
N PGMST ANTSTE	P PRUCSI	CU	DE2 NOTO	K.		A/U
DO STOTETEL						
N PGMST	PROCST	CO	DES			A/0
DO						
HOUT WHEN	TIME	+	DAYS	T0		URGN
MS						
==== >>>>>>>>	>>>>> END	OF SCHE	DULING P.	ARAMETE	RS <<<<<<	<<<<< ======

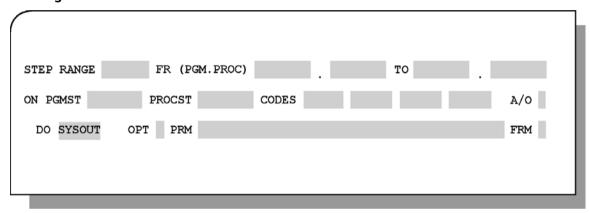
## **DO SYSOUT: Post-Processing Parameter**

Controls handling of job output if the accompanying ON step and code criteria are satisfied.

#### NOTE

DO SYSOUT and SYSOUT statements are similar, but not identical. The differences are outlined below in "Differences between SYSOUT and DO SYSOUT" on page 483."

Figure 203 DO SYSOUT Parameter Format



Optional. Type the word SYSOUT in the DO field and press **Enter**. The DO SYSOUT subparameters are described in Table 181.

Table 181 DO SYSOUT Subparameters

Subparameter	Description
OPT	<ul> <li>SYSOUT option code. Mandatory. Valid values are:</li> <li>F - Copy the job output to file.</li> <li>C - Change the class of the job output.</li> <li>N - Change the destination of the job output.</li> <li>D - Delete (purge) the job output.</li> <li>R - Release the job output.</li> </ul>
PRM	Relevant sysout data. Mandatory and valid only if the specified OPT value is F, C or N. Valid values depend on the OPT value, as follows:  F - File name.  C - New class (1 character). An asterisk (*) indicates the original MSGCLASS of the job.  N - New destination (1 through 8 characters).
FRM	FROM class. Optional. Limits the sysout handling operation to only sysouts from the specified class.  Note: If a FROM class is not specified, all sysout classes are treated as a single, whole unit.

#### **General Information**

The CONTROL-M monitor, unless otherwise instructed, leaves the job sysout in HELD class in the output queue.

The DO SYSOUT parameter is used to request additional handling of these held sysouts when the accompanying ON criteria are satisfied.

The CONTROL-M monitor sends all sysout handling requests to JES, which processes the instructions. If, however, the copying of sysouts to a file is requested (option F), CONTROL-M requests the sysouts from JES and then CONTROL-M directly writes the sysouts to the file.

Since only one SYSOUT statement can be defined in a job scheduling definition, DO SYSOUT statements can be used, as follows, to specify additional sysout handling instructions when the job ends OK:

- To define DO SYSOUT statements that operate like a SYSOUT statement (that is, that operate only when the job ends OK), define their accompanying ON statement with PGMST value ANYSTEP and code value OK.
- The interrelationship between multiple sysout operations (by SYSOUT and DO SYSOUT statements) is described in "Multiple SYSOUT Operations" on page 480.

## **SYSOUT Handling Operations**

The SYSOUTs that are affected by sysout handling operations are determined by whether the SYSOUTs are under JES2 or JES3, as described in Table 182.

Table 182 Varying Effect of SYSOUT Handling Operations

Operation	Effect
Under JES2:	Operations are performed on all of the held SYSOUTs, or held portions of SYSOUTs, of the job, unless otherwise restricted to a specific FROM class by the FRM subparameter.
Under JES3:	Operations are performed only on the SYSOUTs of the job in the CONTROL-M held class, as specified in the CONTROL-M installation HLDCLAS parameter.

SYSOUT handling operations are listed below:

■ Copying SYSOUTs to a file (OPT=F)

The SYSOUTs of the job are copied (not moved) to the file specified in the data subparameter.

The file name specified in the data subparameter can contain AutoEdit System variables, and user-defined AutoEdit variables that are defined in the job scheduling definition or the IOA Global Variable database, or are loaded into AutoEdit cache. If the AutoEdit variables cannot be resolved, the sysout is not copied.

CONTROL-M allocates the file with DISP=(NEW,CATLG,DELETE) using the unit and space attributes specified in the CONTROL-M installation parameters. While the block size (BLKSIZE) is automatically calculated by CONTROL-M, the logical record length (LRECL) is copied from the input SYSOUT file. The maximum LRECL allowed is 256 characters.

SYSOUTs can be archived by copying them. However, to reduce overhead, this method is recommended only for small sysouts.

■ Deleting SYSOUTs (OPT=D)

The SYSOUTs of the job are deleted (purged) from the output queue.

#### NOTE



This operation works on all SYSOUTs under JES2 or JES3 (regardless of held status or class) unless otherwise restricted by the FRM subparameter.

■ Releasing SYSOUTs (OPT=R)

The SYSOUTs of the job are released for printing.

■ Changing the class of SYSOUTs (OPT=C)

The SYSOUTs of the job are changed to the output class specified in the data subparameter. Ensure that you specify a meaningful target output class.

Note the following points:

 Changing a SYSOUT class to a non-held class does not release the SYSOUT because the SYSOUT attributes do not change (due to JES logic).

To ensure that the SYSOUT is released, use DO SYSOUT statements to release the SYSOUT after changing its class. For example: DO SYSOUT OPT C PRM R FRM A

DO SYSOUT OPT R PRM FRM A

- Changing a SYSOUT class to a dummy class does not purge the SYSOUT because the SYSOUT attributes do not change (due to JES logic).
- To save the original MSGCLASS of a job and to restore it at output processing time, specify a data value of *. The SYSOUTs are changed to the original class of the job.
- Moving SYSOUTs to a new destination (OPT=N)

The SYSOUTs of the job are moved to the output destination specified in the data subparameter. Ensure that you specify a meaningful target output destination.

## **Multiple SYSOUT Operations**

If multiple DO SYSOUT (or SYSOUT/DO SYSOUT) operations are not specified for the same FROM class, the order in which the operations are performed is not significant.

However, if different DO SYSOUT (or SYSOUT/DO SYSOUT) operations affect the same FROM class, or if multiple operations are specified without a FROM class, the order and method of implementation is significant.

CONTROL-M merges different operations for the same FROM class into a combined instruction to JES. Likewise, CONTROL-M merges different operations without a FROM class into a combined instruction to JES.

Operations without a specified FROM class treat the entire held sysout as a whole unit, and are therefore not merged with sysout handling requests for a specific FROM class.

JES does not necessarily process multiple sysout handling instructions in the order they are issued by CONTROL-M. Therefore, the processing results can vary if the merged instructions to JES include both FRM equals a specified class and FRM equals blank.

BMC Software therefore recommends that you do not include in a job scheduling definition both "FROM class" and "no FROM class" sysout handling instructions that become operational under the same situations.

When CONTROL-M merges a set of operations into a combined instruction, some operations override or cancel other operations, and some operations are performed along with other operations. This is described below.

## **Operation Merging and Performance**

CONTROL-M performs all copy to file operations (option F) first.

After performing all copy to file operations, CONTROL-M merges all operations performed on a specific FROM class.

After merging operations on specific FROM classes, CONTROL-M merges the operations performed on the sysout as a whole (where the FRM subparameter is set to blank).

CONTROL-M then passes the merged sets of instructions to JES for processing.

The resulting combination of operations can vary depending on whether the operation that was merged with a DO SYSOUT operation is a SYSOUT operation or another DO SYSOUT operation.

Generally, DO SYSOUT operations override, or are performed along with, SYSOUT statements.

The following chart and the accompanying numbered explanations indicate the result of merging multiple DO SYSOUT statements. Note the following points about the chart:

- Operations are indicated by their symbols (F,D,R,C,N), at the top and side of the chart. The operations at the top of the chart represent DO SYSOUT operations. The operations at the side of the chart represent SYSOUT or DO SYSOUT operations.
- Merging and processing operations are grouped, and explained, based on operation type. Groups are delimited by lines, and are numbered (from 1 through 4). Within each group, operations are delimited by periods. Explanations of each group are provided, by number, following the chart.
- The handling of the combination of operations is generally reflected in the chart by a single operation code (such as D) or pair of operation codes, such as FR. In some cases, the operations are merged. This is indicated by the word "merged." In some cases, handling depends on whether the combination consists of both a SYSOUT and a DO SYSOUT statement, or multiple DO SYSOUT statements (that is, without a SYSOUT statement). This is indicated by an asterisk (*). These are all explained in the numbered explanations that follow the chart.

#### NOTE -



For information about merging a SYSOUT and a DO SYSOUT statement, see "Operation Merging and Performance" on page 481.

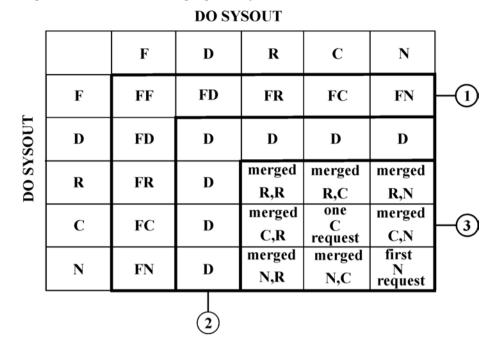


Figure 204 Effect of Merging Multiple SYSOUT Statements.

The order of precedence in which CONTROL-M processes and merges operations is as follows:

#### 1. DO SYSOUT=F

Copy to file operations are performed first (directly by CONTROL-M) for DO SYSOUT statements, regardless of whether FROM class is specified. Other operations are then performed.

#### 2. DO SYSOUT=D (Delete)

This operation supersedes all other DO SYSOUT operations (except copy to file operations described above). Superseded operations are ignored, that is, they are not performed.

#### 3. DO SYSOUT combinations of R, C and N

In general, combinations of R, C, and N requests are merged, that is, they are all performed. The exceptional cases indicated in the chart are described below:

- If multiple C requests come from DO SYSOUT statements, perform only one of the requests. Normally, do not specify this combination.
- If multiple N requests come from DO SYSOUT statements, perform the request that occurs first.

### Differences between SYSOUT and DO SYSOUT

SYSOUT and DO SYSOUT statements have the following differences:

- The SYSOUT statement is applied only if the job ends OK. DO SYSOUT statements are associated with accompanying ON statements and are applied only if the accompanying ON step and code criteria are satisfied.
- A SYSOUT statement appears in each job scheduling definition. A DO SYSOUT statement is not displayed unless requested. To request a DO SYSOUT statement, type SYSOUT in an empty DO field and press **Enter**.
- Only one SYSOUT statement can be defined in the job scheduling definition. An unlimited number of DO SYSOUT statements can be requested.
- The SYSOUT OP subparameter is equivalent to the DO SYSOUT OPT subparameter.
- The SYSOUT data subparameter is equivalent to the DO SYSOUT PRM subparameter.
- The SYSOUT FROM subparameter is equivalent to the DO SYSOUT FRM subparameter.

## **Examples**

#### **Example 1**

If a job finishes executing OK, delete (purge) the sysout (DO SYSOUT OP D). If the job finishes executing with condition code 0050-0059 in step STEP02, set the end status of the job to OK and release the sysout for printing. If the job abends, move the sysout to class D.

Figure 205 DO SYSOUT Parameter – Example 1

JOB: SACALCO1 LIB CTM.PROD.SCHEDULE COMMAND ===>	TABLE: SALARY SCROLL===> CRSR
+	+
OUT	
AUTO-ARCHIVE Y SYSDB Y MAXDAYS MAXRUNS	
RETENTION: # OF DAYS TO KEEP 030 # OF GENERATIONS TO KEEP	
SYSOUT OP (C,D,F,N,R)	FROM
MAXRERUN RERUNMEM INTERVAL	FROM
STEP RANGE FR (PGM.PROC) . TO	
ON PGMST ANYSTEP PROCST CODES OK	A/0
DO SYSOUT OPT D PRM	FRM
ON PGMST STEP02 PROCST CODES CO05*	A/0
DO OK	
DO SYSOUT OPT R PRM	FRM
DO	
ON PGMST ANYSTEP PROCST CODES U**** S****	A/0
DO SYSOUT OPT C PRM D	FRM
DO	
ON PGMST PROCST CODES	A/0
DO	
SHOUT WHEN TIME + DAYS TO	URGN
MS	
====== >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>	<<<<<<< =====
COMMANDS: EDIT, DOC, PLAN, JOBSTAT	11.17.00

#### **Example 2 – Use of the Sysout Archiving Facility**

The MSGCLASS of the job is X (a held class). Reports are produced in class D. The desired actions are:

- Archive the JCL messages and all the held output in class X, that is, the SYSPRINT data sets, job log, and so on.
- If the job finishes executing OK, release the reports for print and delete the MSGCLASS sysouts.
- If the job finishes executing NOTOK, delete the reports and keep the MSGCLASS (JCL, job log, and so on) output in hold status.

Figure 206 DO SYSOUT Parameter – Example 2

JOB: GPLUPDT1 LIB CTM.PROD.SCHEDULE	TABLE: PRODGPL
COMMAND ===>	SCROLL===> CRSR
+	+
OUT	
OUT	
RETENTION: # OF DAYS TO KEEP 030 # OF GENERATIONS TO KEEP	
AUTO-ARCHIVE Y SYSDB Y MAXDAYS MAXRUNS	EDOM
SYSOUT OP (C,D,F,N,R)	FROM
MAXRERUN RERUNMEM INTERVAL	FRUM
STEP RANGE FR (PGM.PROC) . TO ON PGMST ANYSTEP PROCST CODES *****	A/O
DO SYSOUT OPT F PRM GPL.%%JOBNAME.D%%ODATE.N%%JOBID.T%%T	
DO 313001 OFF F FRM GFE. %%000MAME. D%%000ATE. N%%000DID. 1%%1	INL INI A
ON PGMST ANYSTEP PROCST CODES OK	A/0
DO SYSOUT OPT D PRM	FRM X
DO SYSOUT OPT R PRM	FRM D
D0	5
ON PGMST ANYSTEP PROCST CODES NOTOK	A/0
DO SYSOUT OPT D PRM	FRM D
DO	
ON PGMST PROCST CODES	A/0
DO	
SHOUT WHEN TIME + DAYS TO	URGN
COMMANDS: EDIT, DOC, PLAN, JOBSTAT	11.17.00

Notice the use of the AutoEdit symbols in the name of the file to be archived. The symbol %%JOBNAME, or %%\$MEMNAME if the job name is not known, is replaced with the job name, %%ODATE by the original scheduling date, and so on, producing a file name such as "PRD.PADD0040.D010306.N01342.T170843."

The file can be viewed by using ISPF Browse. A list of the outputs of the job can be produced using ISPF option 3.4. For example, retrieval by the prefix "PRD.PAPD0040.D0103" lists all the names of the sysouts of the job in the month of March 2001. It is possible to browse, edit, and print the desired sysout.

#### - NOTE -

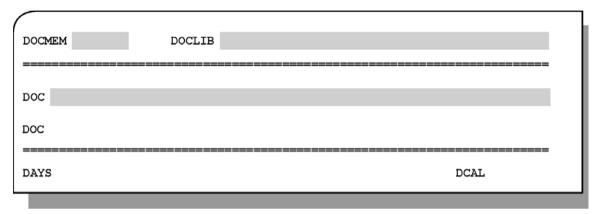


The File operation (sysout archival) is intended for small sysouts (such as JCL, sort messages) and not for large volume reports. When the CONTROL-M monitor is performing file operations, it does not analyze the results of other jobs. Therefore, if large files are archived, production throughput may suffer.

## **DOC: General Job Parameter**

Detailed job documentation. This field can be displayed or hidden by request.

Figure 207 DOC Parameter Format



Optional. Upon filling in a DOC line with text and pressing **Enter**, a new DOC line is opened for specifying additional documentation text.

#### **General Information**

DOC lines are used for specifying job documentation.

Upon entry to the job scheduling definition, DOC lines are displayed only if the value Y was specified in field SHOW JOB DOCUMENTATION in the Scheduling Definition Facility entry panel.

Command DOC can be used in the job scheduling definition to toggle between the display and non display of job documentation.

The information specified in the DOC lines is saved in the member and library specified in the DOCMEM and DOCLIB parameters. This member can also be edited directly by ISPF edit.

When modifying DOC lines in the job scheduling definition, text must be left in at least one DOC line in order to save the modifications. Changes resulting in an empty DOCMEM member are not saved when exiting the job scheduling definition.

For more information regarding job documentation, including the saving of job documentation changes, see "Job Documentation" on page 144.

### **Example**

The steps performed by the L-file backup job are documented in the DOC lines.

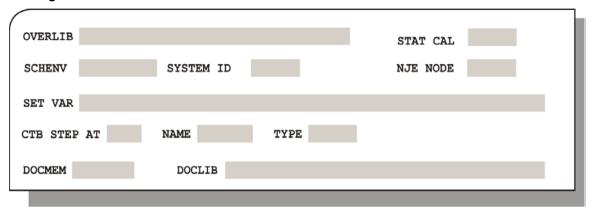
#### Figure 208 DOC Parameter Example

```
JOB: BACKPLO2 LIB CTM.PROD.SCHEDULE
                                                                        TABLE: BACKUP
COMMAND ===>
                                                                        SCROLL==> CRSR
+-----
  MEMNAME BACKPLO2 MEMLIB CTM.PROD.JOBLIB
OWNER M44 TASKTYPE JOB PREVENT-NCT2 Y DFLT N
APPL APPL-L GROUP BKP-PROD-L
DESC DAILY BACKUP OF SPECIAL FILES FROM APPL-L
  OVERLIB CTM.OVER.JOBLIB
                                                                   STAT CAL
  SCHENV
                                   SYSTEM ID
                                                                   NJE NODE
  SET VAR
  CTB STEP AT NAME
                                          TYPE
  DOCMEM BACKPLO2 DOCLIB CTM.PROD.DOC
  DOC THIS JOB BACKS UP "L" FILES. IT PERFORMS THE FOLLOWING STEPS:
  DOC 1: VERIFY SPACE REQUIREMENTS
  DOC 2-5: BACKUP THE FILES
  DOC 6: RECATALOG THE NEW FILES
  DOC 7: PRINT THE SHORT-VERSION LISTING REPORT
  DAYS ALL
                                                                       DCAL
                                                                            AND/OR
  WDAYS
                                                                       WCAL
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
                                                                               11.17.00
```

## **DOCLIB: General Job Parameter**

Name of the library in which the member specified in DOCMEM resides.

Figure 209 DOCLIB Parameter Format



Optional. The DOCLIB parameter identifies a valid data set name of 1 through 44 characters. The default is either defined at time of installation or is blank.

### **General Information**

The library can be any standard partitioned data set. The record length must be 80.

Any number of documentation libraries can be used at a site. However, only one documentation library can be specified in each job scheduling definition.

#### **NOTE**



Users with DOCU/TEXT installed at their sites can specify a DOCU/TEXT library and member with up to 132 characters per line. However, if more than the first 71 characters in a line are used, the line is truncated and Browse mode is forced. Browse mode is also forced if a line contains an unprintable character. Changes to the documentation are not permitted in Browse mode.

### **Example**

Job documentation is written to the PRDKPL01 member in the CTM.PROD.DOC library.

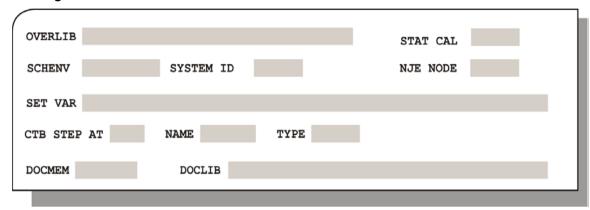
Figure 210 DOCLIB Parameter Example

```
JOB: PRDKPL01 LIB CTM.PROD.SCHEDULE
                                                        TABLE: PRODKPL
COMMAND ===>
                                                        SCROLL===> CRSR
+------
  MEMNAME PRDKPL01 MEMLIB CTM.PROD.JCL
  OWNER SYS1 TASKTYPE JOB PREVENT-NCT2 Y DFLT N
 APPL KPL GROUP PROD-KPL
DESC DAILY PRODUCTION - START OF APPL-PROD-KPL
                                 GROUP PROD-KPL
  OVERLIB
                                                    STAT CAL
  SCHENV
                            SYSTEM ID
                                                    NJE NODE
  SET VAR
  CTB STEP AT
                   NAME
                                 TYPE
  DOCMEM PRDKPL01 DOCLIB CTM.PROD.DOC
  DAYS
        01
                                                       DCAL
                                                            AND/OR
                                                        WCAL
  MONTHS 1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- Y 11- Y 12- Y
  DATES
  CONFCAL
                SHIFT
                         RETRO Y MAXWAIT OO D-CAT
  MINIMUM
                PDS
  DEFINITION ACTIVE FROM
                             UNTIL
         START-DAILY-PROD-KPL ODAT
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
                                                             11.17.00
```

## **DOCMEM: General Job Parameter**

Name of the member that contains job documentation.

Figure 211 DOCMEM Parameter Format



Optional. DOCMEM identifies a valid member name of 1 through 8 characters. The default is either defined during installation or is blank.

### **General Information**

DOCMEM identifies a member that is in the library identified by the DOCLIB parameter. This member is used to save detailed documentation written in the DOC lines of the Job Scheduling Definition screen (or Zoom screen).

When you enter the Job Scheduling Definition screen for the first time, DOCMEM defaults to the value of MEMNAME. You can change this value, but it is recommended that you not do so.

#### NOTE



Users with DOCU/TEXT installed at their sites can specify a DOCU/TEXT library and member with up to 132 characters per line. However, if more than the first 71 characters in a line are used, the line is truncated and Browse mode is forced. Browse mode is also forced if a line contains an unprintable character. Changes to the documentation are not permitted in Browse mode.

## **Example**

Job documentation is written to member PRDKPL01 in the library CTM.PROD.DOC.

#### Figure 212 DOCMEM Parameter Example

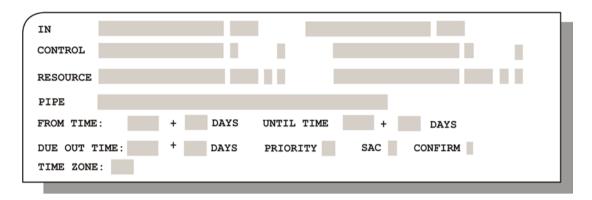
```
JOB: PRDKPL01 LIB CTM.PROD.SCHEDULE
                                                              TABLE: PRODKPL
                                                              SCROLL==> CRSR
+----
 MEMNAME PRDKPLO1 MEMLIB CTM.PROD.JCL

OWNER SYS1 TASKTYPE JOB PREVENT-NCT2 Y DFLT N
 APPI KPI
                                    GROUP PROD-KPL
 DESC DAILY PRODUCTION - START OF APPL-PROD-KPL
 OVERLIB
                                                         STAT CAL
  SCHENV
                               SYSTEM ID
                                                         NJE NODE
  SET VAR
  CTB STEP AT NAME
                                    TYPE
  DOCMEM PRDKPLO1 DOCLIB CTM.PROD.DOC
                                                             DCAL
  DAYS 01
                                                                  AND/OR
  WDAYS
                                                              WCAL
  MONTHS 1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- Y 11- Y 12- Y
  DATES
  CONFCAL
                  SHIFT
                            RETRO Y MAXWAIT OO D-CAT
  MINIMUM
                  PDS
  DEFINITION ACTIVE FROM
                                UNTIL
          START-DAILY-PROD-KPL ODAT
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
                                                                   11.17.00
```

## **DUE OUT: Runtime Scheduling Parameter**

Time and date by which a job or group must finish executing.

Figure 213 DUE OUT Parameter Format



Optional. The format for the DUE OUT TIME parameter is *hhmm*, where:

- *hh* is the hour the job is due out (based on a 24-hour clock)
- *mm* is the minute the job is due out

If the DUE OUT DAYS parameter is entered, the value must be a number between zero and 120. This is the relative number of days by which the job must finish execution. For example, assume that the ODATE is September 5, DUE OUT TIME is 17:00, and DUE OUT DAYS is 2. This means that the job must finish executing by CONTROL-M working date of September 7 at 17:00.

### **General Information**

The DUE OUT parameters are used to specify the date and time by which the job must finish executing.

When specified in a group entity, the DUE OUT parameters are used to specify the date and time by which all the jobs contained in the group must finish executing.

When two jobs with the same priority are available for submission, CONTROL-M submits the job with the earlier DUE OUT date and time first.

When a DUE OUT date and time is specified, the CONTROL-M monitor can calculate a DUE IN date and time for the job.

The DUE IN date and time are the recommended date and time by which the job must be submitted in order to finish executing by the DUE OUT date and time.

If the DUEINCHK parameter in the CTMPARM member in the IOA PARM library has been set to No, the job is always submitted, no matter what values are present in the DUE IN date and time.

If the DUEINCHK parameter in the CTMPARM member in the IOA PARM library has been set to Yes, job submission depends on the DUE IN date and time:

- If the DUE IN date of a job has passed, the job is never submitted.
- If the DUE IN date is not present, and the DUE IN time has passed, the job must wait until the next day to be submitted.

To calculate the DUE IN time, the CONTROL-M monitor subtracts the anticipated elapse time of the job from the DUE OUT time. The anticipated elapse time is the average of the execution times of the job recorded in the CONTROL-M Statistics file.

If DUE OUT date is present, the DUE IN date is calculated as follows:

- The DUE IN date is equal to the DUE OUT date.
- If, when you move forward on the physical clock from the New Day Processing time, you arrive at the DUE OUT TIME before you arrive at the DUE IN time, it means that New Day processing falls between the DUE IN TIME and the DUE OUT TIME. In this case, one day is subtracted from the DUE IN DATE.

If DUE OUT TIME is not specified, the default DUE OUT TIME is the last minute of the working day.

Automatic adjustment of DUE OUT date and time can be requested from the Job Dependency Network screen.

For more information, see "Automatic Job Flow Adjustment" on page 74, and "Job Dependency Network Screen" on page 242. For an explanation of how DUE OUT affects job submission in the QUIESCE command, see the description of setting a planned shutdown time in the *INCONTROL* for z/OS Administrator Guide.

Please note that when specifying a DUE OUT date, the correct MAXWAIT parameter value must be specified. For details, see "MAXWAIT: Basic Scheduling Parameter" on page 519.

### **Example**

Job DISKLOG2 must finish execution by 6:00 A.M., two days from today.

#### Figure 214 DUE OUT Parameter Example

JOB: DISKLOG2 LIB CTM.PROD.SCHEDULE	TABLE: ADABAS
COMMAND ===>	SCROLL===> CRSR
+	+

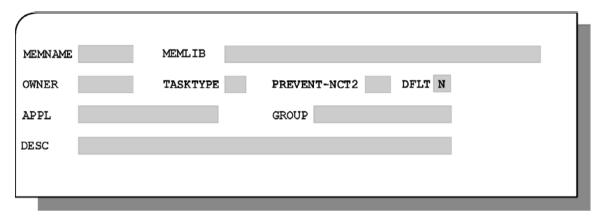
SCHENV	SYSTEM ID	NJE NODE
SET VAR CTB STEP AT NAME	TYPF	
DOCMEM DISKLOG2 DOCLIB	CTM.PROD.DOC	
DAYS		DCAL
		AND/OR O
WDAYS	5 V 6 V 7 V	WCAL
MONTHS 1- Y 2- Y 3- Y 4- Y	5- Y 6- Y /- Y 8	8- Y 9- Y 10- Y 11- Y 12- Y
DATES CONFCAL SHIFT	RFTRO N MAXWAI	T 00 D-CAT
MINIMUM PDS	KLIKO N MAXWAI	1 33 D CAT
DEFINITION ACTIVE FROM	UNTIL	
IN DBA-CLEAN-LOG-2	****	
CONTROL	0001	
RESOURCE TAPE	0001	
FROM TIME + DAYS	S UNTIL TIME	+ DAYS
DUE OUT TIME 0600 + 2 DAYS		SAC CONFIRM
TIME ZONE:		
COMMANDS: EDIT, DOC, PLAN, JO	DBSTAT	17.14.10

I

## **GROUP: General Job Parameter**

Group to which a job belongs.

Figure 215 GROUP Parameter Format



The GROUP parameter identifies a group name of 1 through 20 characters. Only trailing blanks are allowed.

- In a Group Entity, the parameter is mandatory.
  - In jobs in a Group scheduling table, the field is protected and contains the GROUP name specified in the Group Entity.
- By default, the parameter is optional for jobs in regular scheduling tables, but this can be modified in the user profile. The same value does not have to be specified for all jobs in the table.

### **General Information**

The way in which the GROUP parameter is applied depends on the type of scheduling table in which the job scheduling definitions appear:

- In a Group scheduling table, the parameter affects job scheduling as well as the retrieval and display of information.
- In a regular scheduling table, the parameter affects the retrieval and display of information. It does not affect job scheduling.

## **Group Job Scheduling**

When a Group scheduling table is created, a value for the GROUP parameter must be specified in the Group Entity. This value is automatically applied to the GROUP field in all job scheduling definitions in the table.

Jobs in a Group scheduling table cannot be individually ordered. Jobs in this type of table can only be ordered as a group, though they can be individually forced.

Before jobs in the Group scheduling table can be scheduled, a group must be eligible for scheduling, meaning that a set of basic scheduling criteria in the Group Entity must be satisfied.

Basic scheduling criteria, runtime scheduling criteria, and post-processing parameters in the Group Entity apply to all scheduled jobs in the group.

For more information, see "Handling of Job Groups" on page 68 and page 111, and "Scheduling Jobs in Group Scheduling Tables" on page 382.

## **Retrieving and Displaying Information**

Regardless of scheduling table type, the GROUP parameter can be used as a selection criteria that can make retrieval and display of information more efficient.

For example, display of information in the Active Environment screen can be limited to jobs belonging to a specific group.

The group name appears in all important messages relating to the jobs in the group.

#### NOTE



BMC Software recommends the use of the GROUP parameter in all job scheduling definitions to facilitate implementation of CONTROL-M/Enterprise Manager functions. For more information, see the *CONTROL-M/Enterprise Manager User Guide*.

## **Example**

Job OPERCOMP (in a regular scheduling table) belongs to the MAINTENANCE group.

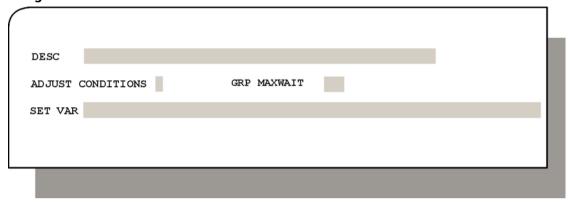
Figure 216 GROUP Parameter Example

```
JOB: OPERCOMP LIB CTM.PROD.SCHEDULE
                                                      TABLE: OPER
COMMAND ===>
                                                     SCROLL===> CRSR
+------
 MEMNAME OPERCOMP MEMLIB CTM.PROD.JCL
 OWNER SYS1 TASKTYPE JOB PREVENT-NCT2 Y DFLT N
 APPL OPER GROUDESC JOB RUN ON THE 1ST OF THE MONTH
                                GROUP MAINTENANCE
  OVERLIB
                                                  STAT CAL
  SCHENV
                           SYSTEM ID
                                                  NJE NODE
  SET VAR
 CTB STEP AT
                  NAME
                                TYPE
 DOCMEM OPERCOMP DOCLIB CTM.PROD.DOC
 DAYS
        01
                                                     DCAL
                                                         AND/OR
                                                     WCAL
 MONTHS 1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- Y 11- Y 12- Y
  DATES
  CONFCAL
               SHIFT
                        RETRO Y MAXWAIT OO D-CAT
 MINIMUM
                PDS
 DEFINITION ACTIVE FROM
                           UNTIL
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
                                                          11.17.00
```

# **GRP MAXWAIT: Basic Scheduling Parameter**

Number of extra days the Group Entity can wait in the Active Jobs file if it does not have an ENDED OK status. This parameter appears in and applies to the Group Entity only.

Figure 217 GRP MAXWAIT Parameter Format



Optional. Valid values are: any 2-digit number in the range of 00-98, or 99.

Table 183 GRP MAXWAIT Parameter Values

Value	Description
00	If the Group Entity did not receive an ENDED OK status on its original scheduling date, it cannot remain in the Active Jobs file beyond its original scheduling date, unless jobs belonging to the Group Entity are still in the Active Jobs file. Default.
nn	Where $nn = 01 - 98$ . If Group Entity did not receive an ENDED OK status on its original scheduling date, it can remain in the Active Jobs file up to $nn$ additional days awaiting that status.
99	Group Entity remains in the Active Jobs file until deleted manually, even if it has an ENDED OK status.

If no value is specified, the default value of 00 is automatically inserted. This default value may be changed by your INCONTROL administrator, by means of Wish WM2367 in member IOADFLT in the IOA IOAENV library.

#### **General Information**

The GRP MAXWAIT parameter enables the Group Entity to remain in the Active Jobs file for the specified number of days beyond the original scheduling date if the Group Entity did not receive an ENDED OK status.

This parameter is relevant only when there are no jobs belonging to the Group Entity in the Active Jobs file. As long as a job belonging to the Group Entity is still in the Active Jobs file, the Group Entity remains in the Active Jobs file regardless of the value in the GRP MAXWAIT field.

For more information, see "MAXWAIT: Basic Scheduling Parameter" on page 519.

## **Example**

If the original scheduling date of the Group Entity has passed, give it an extra three days to receive a status of ENDED OK.

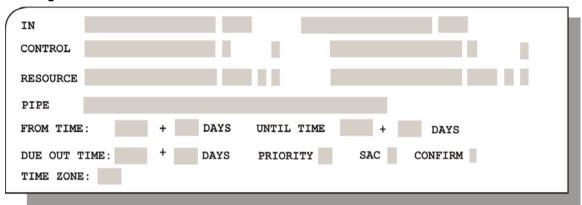
Figure 218 GRP MAXWAIT Parameter Example

```
GRP ACCOUNTS_GROUP CTM.PROD.SCHEDULE(GRP)
COMMAND ===>
                                                              SCROLL===> CRSR
  GROUP ACCOUNTS_GROUP MEMNAME ACCOUNTS
  OWNER NO4B
  APPL
  DESC
  ADJUST CONDITIONS Y
                                GRP MAXWAIT 03
  SET VAR
  DOCMEM ACCOUNTS DOCLIB CTM.PROD.DOC
  SCHEDULE TAG ALL_DAYS
  DAYS ALL
                                                              DCAL
                                                                  AND/OR
                                                              WCAL
  WDAYS
  MONTHS 1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- Y 11- Y 12- Y
  DATES
  CONFCAL SHIFT RETRO N MAXWAIT 00 SCHEDULE TAG ACTIVE FROM UNTIL
  SCHEDULE TAG SUNDAYS
                                                              DCAL
  DAYS
        0.1
                                                                  AND/OR
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
                                                                    16.44.31
```

## **IN: Runtime Scheduling Parameter**

Prerequisite conditions that must be satisfied before the job can run.

Figure 219 IN Parameter Format



Optional. A maximum of two prerequisite conditions can be specified in each standard IN line. One prerequisite condition can be specified in each long IN line. When you specify the second prerequisite condition in a standard IN line, or one prerequisite condition in a long IN line, and press **Enter**, a new IN line is opened for specifying additional prerequisite conditions. For more information, see "Specifying Long IN Condition Names" on page 502.

Each specified prerequisite condition consists of the mandatory subparameters described in Table 184.

Table 184 IN Subparameters (part 1 of 2)

Subparameter	Description
cond_name	User-supplied descriptive name of 1 through 39 characters used to identify the condition. Mandatory.
	<b>Note</b> : A condition name must not begin with the symbols " ", "¬", or "\", and must not contain parentheses (), because each of these characters has a special meaning. For more information, see "Logical Relations between Multiple Conditions" on page 503.
	You can use an AutoEdit variable in a condition name, provided that the AutoEdit variable has a value that is known before the job is ordered.

Table 184 IN Subparameters (part 2 of 2)

Subparameter	Description
dateref	4-character date reference. Mandatory. Valid values are:
	<ul> <li>date – Specific date (in either mmdd or ddmm format, depending on the site standard).</li> </ul>
	■ ODAT – Resolves to the original scheduling date. Default.
	■ +nnn - Resolves at job order time to ODATE+nnn calendar days. nnn is three digits (000-999).
	■ -nnn – Resolves at job order time to ODATE-nnn calendar days. nnn is three digits (000-999).
	<b>Note:</b> -001 is not necessarily the same as PREV, because PREV is based on job scheduling criteria, while - <i>nnn</i> is based on calendar days.
	<ul> <li>PREV – Resolves to the previous date on which the job ought to have been scheduled, according to its basic scheduling criteria, or ODATE-1 for a forced job.</li> </ul>
	Note: for Group Scheduled Jobs: If the value of the SCHEDULE TAG parameter has been set to * (asterisk), PREV is resolved to the nearest previous date that satisfies one or more Schedule Tags in the Group entity.
	■ STAT – Static. Indicates that the condition, such as IMS-ACTIVE, is not date-dependent.
	Note: Before STAT was introduced, date 0101 was recommended to be used in conditions that were not date-dependent. Unlike 0101, STAT is not a date, and it operates differently. Always use STAT when defining conditions that are not date-dependent.
	■ **** – Any scheduling date
	■ \$\$\$\$ – Any scheduling date
	<b>Note:</b> If a date reference is not specified, value ODAT is automatically inserted when you press <b>Enter</b> .

## **General Information**

A job cannot be submitted unless all the prerequisite condition criteria specified in the IN statements have been satisfied.

Prerequisite conditions are usually used to establish job dependencies or to ensure manual intervention when required:

■ To establish job dependency, define a prerequisite condition in an OUT or DO COND statement in the job that must run first, and in an IN statement in the job that must run afterwards.

The job containing a prerequisite condition in its IN statement is not submitted unless that prerequisite condition has been added manually or by the job containing the OUT or DO COND statement.

- An OUT statement adds the prerequisite condition if the job ends OK.
- The DO COND statement adds the prerequisite condition if the step and code event criteria specified in the accompanying ON statement are satisfied.
- If the IN prerequisite condition can only be satisfied by manual intervention (for example, TAPE1-ARRIVED is set by the operator after an external tape arrives on-site), performance of the required manual intervention before job submission can be ensured.

OUT and DO COND statements can also be used to delete prerequisite conditions that are no longer needed. If an IN prerequisite condition for a job is not an IN prerequisite condition for any other job, you can use the OUT statement of the job to delete the prerequisite condition after the job ends OK.

The following are examples of prerequisite conditions:

IMS-ACTIVE
JOB_PAYCALC_ENDED_OK
TAPE1_LOADED

All prerequisite conditions are created with a date reference. When specifying a prerequisite condition as an IN condition, you must specify the date for the condition. Only a prerequisite condition with the specified date can satisfy the IN requirement.

For more information regarding prerequisite conditions, see "OUT: Post-Processing Parameter" on page 562, "ON Statements: Post-Processing Parameter" on page 537, and "DO COND: Post-Processing Parameter" on page 436, and see "Prerequisite Conditions" on page 69

## **Specifying Long IN Condition Names**

Regular prerequisite conditions are not more than 20 characters in length. If you want to specify a longer condition name, up to 39 characters in length, enter the string **LONG** in the date reference field of an empty IN condition line. An (L) appears at the beginning of the line. If the field already contains data, entering the string **LONG** will open a new long IN condition parameter, with (L) appearing at the beginning of the line. You can now insert a long condition name, as illustrated in Figure 220 on page 503.

Specify **SHRT** in the date reference field to revert back to condition names of standard length.

#### NOTE



Long condition names cannot be used in CMEM rule definitions.

#### Figure 220 Long IN Condition

```
JOB: J13 LIB CTMP.V610.SCHEDULE
                                                                                                                                                                                                                                                                        TABLE: REV1
 COMMAND ===>
                                                                                                                                                                                                                                                                      SCROLL===> CRSR
                                           CTMLDNRS-NMIS-OK ODAT CTMLDNRS-NMIS-OK1 ODAT
                                (L) THIS-IS-A-LONG-IN-CONDITION-NAMEXXXXXXX ODAT
         RESOURCE
         FROM TIME + DAYS UNTIL TIME + DAYS
DUE OUT TIME + DAYS
THE TAKE TO BE TO
                                                                             + DAYS PRIORITY SAC CONFIRM
         TIME ZONE:
         OUT J1-ENDED ODAT +
AUTO-ARCHIVE Y SYSDB Y MAXDAYS MAXRUNS
ACTION # OF DAYS TO KEEP # OF GENERATIONS TO KEEP
         SYSOUT OP (C,D,F,N,R)
MAXRERUN RERUNMEM
                                                                                                                                                                                                                                                                                                           FROM
         STEP RANGE
                                                                                                                                                                                                                                     INTERVAL
                                                                                                                                                                                                                                                                                              FROM
         STEP RANGE FR (PGM.PROC)
ON PGMST PROCST
                                                                                                                                                                                                                                                  T0
                                                                                                                                                                      CODES
                                                                                                                                                                                                                                                                                                          A/0
                D0
         SHOUT WHEN LATE TIME 1300 +
                                                                                                                                                                      DAYS
                                                                                                                                                                                                      TO TS0-N88
                                                                                                                                                                                                                                                                                              URGN R
                MS BBB
          SHOUT WHEN
                                                                                                                                                                      DAYS
                                                                                                                                                                                                                                                                                                URGN
      COMMANDS: EDIT, DOC, PLAN, JOBSTAT
                                                                                                                                                                                                                                                                                                       09.06.50
```

## **Logical Relations between Multiple Conditions**

The IN condition parameter differs from other parameters that may be defined more than once.

Where there are multiple IN condition definitions, they are not independent parameters, as might at first appear. CONTROL-M takes them together and treats them as a logical expression consisting of a series of connected terms, which appear as condition names.

CONTROL-M resolves every such condition to a value of "True" or "False," and then evaluates the whole expression, using the logical operators which may have been specified as part of each condition name. The run-time criteria for prerequisite IN conditions are only satisfied if the overall value of the expression is calculated as "True". A condition name is evaluated as "True" if the name of that condition appears in the IOA Conditions file.

Conditions may be added to or deleted from the IOA Conditions file automatically or manually. Some typical means of adding and/or deleting conditions are:

- the CONTROL-M Monitor, by means of OUT or DO COND statements
- the IOA Conditions/Resources screen
- the IOA Manual Conditions screen
- the IOACND or IOACLND batch utilities

The following types of logical operator can be used to connect condition names:

- unitary
- binary
- group

#### NOTE



These operators are not referred to as "Boolean", because the rules of these operators do not follow formal Boolean logic, as shown in the following paragraphs. Logical operators are the first physical characters in condition names, but they are not part of the condition name.

## **Operators: Unitary**

The logical NOT is the only unitary operator. It is represented in the condition name by the symbol  $\neg$  (Hex 5f) or  $\setminus$  (Hex e0). Conditions that have this type of symbol associated with them are called "inverted" conditions. An inverted condition is only "True" if that condition does not exist on the IOA Conditions File.

## **Operators: Binary**

The following are the binary operators:

- logical AND
   This is implicit, and has no explicit representation in the condition name.
- logical OR, represented by the symbol | (Hex 4f)

Where an expression contains conditions connected by an AND operator, both must be present in the IOA Conditions File for the expression to be "True".

An expression that contains conditions connected by an OR operator is "True" if either expression is present in the IOA Conditions File.

Because logical OR operators are expressed as part of the condition name,

 all conditions connected by the logical OR must specify the OR symbol in their condition name This means that, for example, expressions of the form

A | B

must not be specified because their meaning is unclear, even though they will not be syntactically rejected. In reality, because condition name A does not have an OR symbol attached to it, no logical OR connection exists between A and B, and the OR symbol attached to condition name B is ignored. The correct way to specify "Condition A OR Condition B" is

(|A||B)

 all condition names that specify an OR symbol are processed first, before those specifying an AND symbol

This has the effect of creating implicit parentheses among the terms of the expression (explained under "Operators: Group" below); the terms of the expression may also be rearranged.

For example, the expression

is processed as if the expression had been

## **Operators: Group**

The group operator is the pair of parentheses, Open, represented by the symbol (, and Close, represented by the symbol). These must always appear in matched pairs. Parentheses affect the order in which the other logical operators are applied to the terms of the expression. Always specify parentheses when coding an expression that contains different logical operators, to ensure that the terms are combined in the way you want.

Various combinations of logical operators are permitted, subject to the following limitations:

- only one level of parenthesis nesting is allowed
- double NOT operators are not supported
- an open parenthesis preceded by a NOT operator is not allowed

As in standard logic (de Morgan's Rules), the following expressions express logical equivalence:

$$A \quad (\mid B \mid C) \qquad \text{and} \qquad \mid (A \mid B) \mid \mid (A \mid C)$$
 
$$\mid A \mid \mid (B \mid C) \qquad \text{and} \qquad (\mid A \mid \mid B) \quad (\mid A \mid \mid C)$$

A  $\neg$  A is always "False".

### **Example**

A job containing this combination of IN conditions will be selected for execution when the following statements are both "True".

- B exists and C does not exist
- A exists, or D does not exist, or (E does not exist and F exists)

# **IN Parameter Examples**

The following are examples of the IN parameter.

#### **Example 1**

Schedule the job that produces the salary statistics report for top management when the set of jobs that calculates the salaries finishes OK.

When the set of jobs that calculates the salaries finishes OK, it creates the prerequisite condition SALARY-OK.

The report is produced twice a month, for the 1st and for the 15th. The report for the 15th is produced only if the prerequisite condition for the 15th, SALARY-OK, exists, signifying that the salary job for the 15th ended OK. The existence of the prerequisite condition for the 1st, SALARY-OK, does not cause submission of the report for the 15th.

Figure 221 IN Parameter – Example 1

```
JOB: EBDRPT1A LIB CTM.PROD.SCHEDULE
                                                     TABLE: EBDPROD
COMMAND ===>
                                                      SCROLL ===> CRSR
                   GROUP EBD-PRODUCTION
  DESC EBD PRODUCTION SALARY REPORTS
  OVERLIB
                                                   STAT CAL
  SCHENV
                           SYSTEM ID
                                                   NJE NODE
  SET VAR
  CTB STEP AT NAME
                                TYPF
  DOCMEM EBDRPT1A DOCLIB CTM.PROD.DOC
  DAYS 01.15
                                                       DCAL
                                                          AND/OR
  WDAYS
                                                       WCAL
  MONTHS 1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- Y 11- Y 12- Y
  DATES
                SHIFT RETRO Y MAXWAIT 06 D-CAT
  CONFCAL
                PDS
  MINIMIM
  DEFINITION ACTIVE FROM
                           UNTIL
         SALARY-OK
  CONTROL
  RESOURCE
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
                                                   11.17.00
```

#### **Example 2**

This example is similar to Example 1. A monthly total report must be produced based on data from the last two runs, and the job must run when IMS is active.

Figure 222 IN Parameter – Example 2

```
JOB: EBDRPT1A LIB CTM.PROD.SCHEDULE
                                                         TABLE: EBDPROD
COMMAND ===>
                                                         SCROLL===> CRSR
 DESC EBD PRODUCTION REPORTS
 OVERLIB
                                                     STAT CAL
  SCHENV
                            SYSTEM ID
                                                     NJE NODE
  SET VAR
 CTB STEP AT NAME
                                TYPF
  DOCMEM EBDRPT1A DOCLIB CTM.PROD.DOC
  DAYS
        01,15
                                                         DCAL
                                                             AND/OR
  WDAYS
                                                         WCAL
 MONTHS 1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- Y 11- Y 12- Y
  DATES
  CONFCAL
                SHIFT RETRO Y MAXWAIT 06 D-CAT
 MINIMUM
                PDS
  DEFINITION ACTIVE FROM
                             UNTIL
         SALARY-OK
                            ODAT SALARY-OK
                                                       PREV
          IMS-ACTIVE
                            STAT
  CONTROL
  RESOURCE
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
                                                               11.17.00
```

Prerequisite condition IMS-ACTIVE is based on a static condition that exists only when IMS is active. IMS itself can be monitored by CONTROL-M. When IMS is not active, CONTROL-M deletes the prerequisite condition IMS-ACTIVE, thus preventing abends of jobs that depend on IMS.

#### **Example 3**

Assume that there is a group of jobs that runs every day of the week except Saturday and Sunday. It is very important that some of the jobs scheduled for the different days of the week do not run simultaneously. The order of these jobs must be maintained even if there are delays.

Figure 223 IN Parameter – Example 3

```
JOB: FBDUPDT2 LIB CTM.PROD.SCHEDULE
                                                         TABLE: FBDPROD
COMMAND ===>
                                                         SCROLL===> CRSR
  APPL EBD GROUP EBD-PRODUCTION
  DESC EBD PRODUCTION UPDATE
  OVERLIB
                                                       STAT CAL
  SCHENV
                            SYSTEM ID
                                                       NJE NODE
  SET VAR
  CTB STEP AT NAME TYPE
DOCMEM EBDUPDT2 DOCLIB CTM.PROD.DOC
  DAYS
                                                          DCAL
                                                               AND/OR
  WDAYS 2,3,4,5,6
                                                           WCAL
  MONTHS 1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- Y 11- Y 12- Y
  DATES
  CONFCAL SHIFT MINIMUM PDS
                           RETRO Y MAXWAIT 08 D-CAT
  DEFINITION ACTIVE FROM
                              UNTIL
         DEPOSITS
                            PREV
  CONTROL
  RESOURCE
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
                                                                 11.17.00
```

The job is submitted only if the prerequisite condition DEPOSITS of the previous schedule date exists. The prerequisite condition DEPOSITS is created only after the group of jobs called DEPOSITS finishes.

#### **Example 4**

This report must run after the database has been updated by either of two jobs, EBDUPDT2 or EBDUPDT3, but only if IMS is active.

Figure 224 IN Parameter – Example 4

```
JOB: EBDRPT6C LIB CTM.PROD.SCHEDULE
                                                              TABLE: EBDPROD
COMMAND ===>
                                                              SCROLL ===> CRSR
  DESC EBD PRODUCTION DATABASE REPORTS
  OVERLIB
                                                          STAT CAL
  SCHENV
                              SYSTEM ID
                                                          NJE NODE
  SET VAR
 SET VAR
CTB STEP AT NAME TYPE
DOCMEM EBDRPT6C DOCLIB CTM.PROD.DOC
  DAYS 01,15
                                                              DCAL
                                                                  AND/OR
                                                              WCAL
  MONTHS 1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- Y 11- Y 12- Y
  DATES
                  SHIFT RETRO Y MAXWAIT 04 D-CAT
  CONFCAL
                  PDS
  MINIMUM
  DEFINITION ACTIVE FROM UNTIL
          |EBD-EBDUPDT2-ENDED ODAT |EBD-EBDUPDT3-ENDED ODAT
          IMS-ACTIVE STAT
  CONTROL
  RESOURCE
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
                                                                    11.17.00
```

This job is submitted only if IMS is active and if job EBDUPDT2 (or EBDUPDT3) finished executing.

### **Example 5**

Use of parentheses in the IN conditions is demonstrated in the following example. Job EDBCLEAN requires that two conditions be satisfied before submission. The first must be either condition CICSP1-IS-UP or condition CICSP2-IS-UP. The second must be either condition OPR-CLEAN-REQUEST or condition SYS-CLEAN-REQUEST.

Figure 225 IN Parameter – Example 5

JOB: EBDC COMMAND ==		HEDULE	TABLE: EBDPROD SCROLL===> CRSR
OVERLIB SCHENV SET VAR	CTM.OVER.JOBLIB	SYSTEM ID	STAT CAL NJE NODE
	P AT NAME EBDCLEAN DOCLIB	TYPE	
DAYS	ALL		DCAL AND/OR
WDAYS MONTHS DATES	1- Y 2- Y 3- Y 4- Y	5- Y 6- Y 7- Y	WCAL 8- Y 9- Y 10- Y 11- Y 12- Y
CONFCAL MINIMUM DEFINIT		RETRO N MAXWAI UNTIL	T 00 D-CAT
I N	(CICSP1-IS-UP (OPR-CLEAN-REQUEST		SP2-IS-UP) 0101 -CLEAN-REQUEST) ODAT
	E INIT : EDIT, DOC, PLAN, J		T 0001 11.17.00

### **Example 6**

The following example provides a further explanation of the concept of the schedule date reference:

Figure 226 IN Parameter – Example 6

```
MEMNAME EBDRPT6D

MEMLIB EBD.PROD.JOB

DAYS 01,15,20

MONTHS 1- N 2- N 3- N 4- N 5- N 6- N 7- Y 8- N 9- Y 10- N 11- N 12- N

IN EBD-REPORTS-READY ****
```

Today is the 15th September. The date reference values resolved in this job are written in mmdd date format:

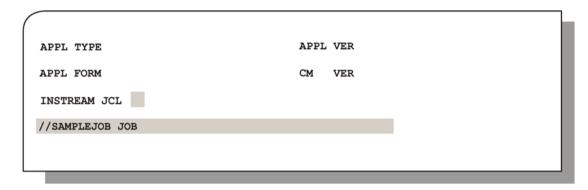
**Table 185 Date Reference Values – Example 6** 

Subparameter	Value
ODAT	0915
PREV	0901
***	Any date reference.

# **INSTREAM JCL: General Job Parameter**

Whether CONTROL-M for z/OS submits a JCL stream defined within the job scheduling definition, overriding the JCL in the member identified in the MEMLIB parameter and the OVERLIB parameter (if specified).

Figure 227 INSTREAM JCL parameter format



Optional. Valid values for this parameter are:

- Y Submit the defined JCL statements as the JCL for the job
- N Use the JCL in the member identified in the MEMLIB parameter (default)

Under the INSTREAM JCL field is an empty line in which you can type a JCL statement. No JCL statement can contain more than 72 characters, including spaces. When you press **Enter**, a new line is opened in which another JCL statement can be typed.

You can enter up to 50 lines of JCL statements. The contents of these lines can be edited subsequently.

When the INSTREAM JCL parameter is set to Y, the library and member specified in the job definition are ignored.

## **General Information**

Prior to version 6.2.00 of CONTROL-M for z/OS, user-defined JCL that was to be run in the course of the execution of a job had to be defined in a JCL library.

The INSTREAM JCL parameter enables you to include JCL that is to be run in the course of the execution of a job within the definition of the job itself.

When the job is ordered or forced, any JCL defined using the INSTREAM JCL parameter resides in the Active Jobs file. The content of the JCL statements can then be modified by means of the Zoom and Save commands.

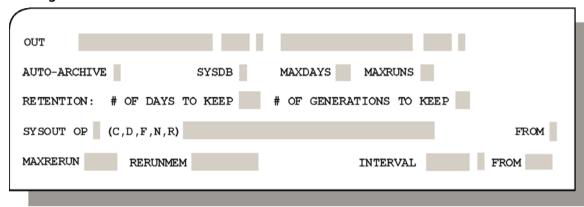
JCL that has been defined using the INSTREAM JCL parameter is processed by submit-related user exits, such as CTMX002, in the same way as JCL retrieved through the MEMLIB and MEMNAME parameters.

# **INTERVAL: Post-Processing Parameter**

Minimum time to wait between automatic reruns or cyclic runs of a job.

A related topic, cyclic jobs, is discussed in "TASKTYPE: General Job Parameter" on page 654.

Figure 228 INTERVAL Parameter Format



Optional. INTERVAL consists of the subparameters described in Table 186.

Table 186 INTERVAL Subparameters (part 1 of 2)

Subparameter	Description
interval_number	A number from 0 through 64800, depending on the value entered in the <i>interval_type</i> field, specifying the minimum time to wait between reruns or cyclic runs. Leading zeros are not required. Mandatory.
	Default: 00000, indicating that there is no minimum time interval between runs.

Table 186 INTERVAL Subparameters (part 2 of 2)

Subparameter	Description
interval_type	A single character describing the type of data specified in the INTERVAL field. Valid values are:
	<ul> <li>D (Days) - Maximum INTERVAL value is 45</li> <li>H (Hours) - Maximum INTERVAL value is 1080</li> <li>M (Minutes) - Maximum INTERVAL value is 64800. Default.</li> </ul>
FROM	Determinant of when the time to wait between reruns or cyclic runs of a job begins. Valid values are:
	■ STRT – Begin measuring the interval before the next cycle of the job from the actual start of the current job run.
	■ END – Begin measuring the interval before the next cycle of the job from the end of the current job run. Default.
	■ TRGT – Begin measuring the interval before the next cycle of the job from when the current job run is scheduled.

### **General Information**

The INTERVAL parameter defines a minimum interval between automatic reruns or cyclic runs of the same job.

Once the job has run, the CONTROL-M Monitor does not rerun or resubmit the job unless both the following conditions are satisfied:

- the specified time has passed
- all runtime submission criteria, such as resources, conditions, and so on, are satisfied

The FROM subparameter specifies the point from which the interval is measured. The values set for this subparameter have the following effects:

- If STRT is specified, the interval is measured from the start time of the previous run.
- If END is specified, the interval is measured from the time the previous run ended.
- If TRGT is specified, the interval is measured from the scheduling time of the current job run.

If no value was specified in the TIME FROM parameter, the interval is measured from the time the CONTROL-M Monitor scheduled the current job run.

For more information about the TIME FROM parameter, see "TIME + DAYS: Runtime Scheduling Parameter" on page 659.

A job's INTERVAL parameter is ignored when performing a Rerun (R) command for a job in screen 3.

# **Example**

A backup for an ADABAS database failed because the database was being used by another user. Backups are tried every 15 minutes after the job ends, to a maximum of nine attempts.

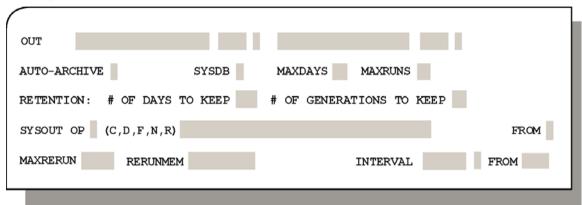
Figure 229 INTERVAL Parameter Example

```
JOB: ADBBKPS LIB CTM.PROD.SCHEDULE
                                                  TABLE: ADABAS
COMMAND ===>
                                                  SCROLL===> CRSR
                    SYSDB Y
                                 MAXDAYS
 RETENTION: # OF DAYS TO KEEP 030 # OF GENERATIONS TO KEEP
 SYSOUT OP (C,D,F,N,R)
                                     INTERVAL 0015 M FROM END
 MAXRERUN 9 RERUNMEM
 STEP RANGE
               FR (PGM.PROC)
                                          T0
 ON PGMST BACKUP PROCST
                           CODES U0034
                                                        A/0
   DO RERUN
   D0
 ON PGMST
           PROCST
                            CODES
                                                        A/0
   D0
 SHOUT WHEN
                 TIME
                               DAYS
                                      T0
                                                      URGN
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
                                                       11.17.00
```

# **MAXRERUN: Post-Processing Parameter**

Maximum number of automatic reruns to be performed for the job. Called RERUN – MAXRERUN prior to version 6.0.00.

Figure 230 MAXRERUN Parameter Format



Optional. For non-cyclic jobs, valid values are 0 through 255. For cyclic jobs, valid values are 0 through 9999. Default: 0 (no automatic reruns or cyclic iterations).

## **General Information**

When a job is first run, the MAXRERUN field in the Active environment, that is, in the Zoom screen, contains the same value as the MAXRERUN parameter in the job scheduling definition. However, in the Active environment MAXRERUN works as a "reverse-counter" of automatic reruns. Each time the job is automatically rerun, the value is decreased by one until the field contains a value of zero.

The automatic rerun process works as follows:

- 1. The CONTROL-M monitor determines that automatic rerun is possible only if the job ENDS NOTOK and a specified DO RERUN statement is activated during post-processing. If the monitor determines that automatic rerun is possible, it sets the status of the job to ENDED NOTOK RERUN NEEDED.
- 2. The monitor then checks the value of MAXRERUN in the Active environment. If the value is zero, automatic rerun is not possible and the job is not submitted for rerun. If the value is greater than zero, rerun is possible and the monitor submits the job for rerun when all runtime criteria are satisfied. Runtime criteria include not only criteria in the Runtime Scheduling parameters, but also the INTERVAL parameter, which specifies the minimum allowable interval between runs of the same job.

3. The JCL for the rerun job is taken from the member specified in the RERUNMEM parameter. If no RERUNMEM value is specified, the JCL for the rerun is taken from the regular JCL member of the job that is specified in the MEMNAME parameter.

MAXRERUN applies only to automatic reruns. The MAXRERUN counter is not affected by reruns performed manually using the Rerun option in the Active Environment screen.

If a job is defined as cyclic by setting the TASKTYPE parameter to CYC, the MAXRERUN parameter can be used to specify the number of iterations. This number excludes the initial run of the job.

# **Examples**

### **Example 1**

A tape I/O error occurred. Try two more times. If there are two more failures, terminate:

MAXRERUN 2 RERUNMEM		INTERVAL	0015	М	FROM END
ON PGMST STEP01 PROCST	CODES	S613			
DO RERUN					

## **Example 2**

When a job abends for any reason, try to restart it two more times (at the abended step).

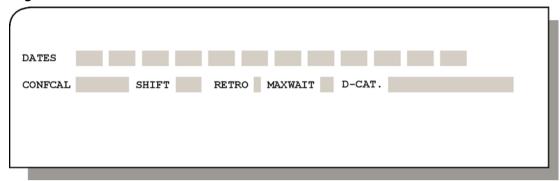
Figure 231 MAXRERUN Parameter Example 2

OUT PRDKPL	01 - FNDFD - OK		 ODAT +			
AUTO-ARCHIVE Y				(YS 1	1AXRUNS	
RETENTION: # 0	F DAYS TO KEEP	030 1	F OF GENER	RATIONS TO	) KEEP	
SYSOUT OP (C,	D,F,N,R)					FROM
MAXRERUN 2 RE						M FROM END
STEP RANGE						•
ON PGMST ANYSTE				U****	C2000	
DO IFRERUN FF DO RERUN DO	ROM \$ABEND .		T0	•		CONFIRM N
ON PGMST DO	PROCST	CC	DES			A/0
SHOUT WHEN MS	TIME	+	DAYS	T0		URGN
>>>>>>>>>>>	>>>>> END	OF SCHE	EDULING PA	RAMETERS	<<<<<<	

# **MAXWAIT: Basic Scheduling Parameter**

Number of extra days the job can wait in the Active Jobs file for submission.

Figure 232 MAXWAIT Parameter Format



Optional. Valid values are: any 2-digit number in the range from 00 through 98, or 99.

Table 187 MAXWAIT Parameter Values

Value	Description
00	Job is not executed if it did not execute on the original scheduling date. Default.
nn	Where $nn = 01 - 98$ . If the job did not execute on its original scheduling date, it is given an additional number of days to execute. It can remain in the Active Jobs file up to nn days awaiting execution.
99	Job remains in the Active Jobs file until deleted manually, even if the job finished executing.

If no value is specified, the default value of 00 is automatically inserted. This default value may be changed by your INCONTROL administrator, by means of Wish WM2367 in the IOADFLT member in the IOA IOAENV library.

## **General Information**

The MAXWAIT parameter is used to overcome the problem of delays in production. A job that is scheduled for execution on a specific day does not always get executed that same day. This can be due to a number of reasons, such as hardware failure or a heavy production workload. Therefore, it may be desirable to specify an additional number of days that the job must remain in the Active Jobs file awaiting execution.

When a job cannot be submitted for execution within the specified time limits, an appropriate message is written to the IOA Log file, and the job is deleted from the Active Jobs file.

Jobs scheduled as a result of a Y value in the RETRO parameter are always given at least one day within which to execute, even if the MAXWAIT parameter indicates that they must no longer be in the Active Jobs file. This occurs when the current working date exceeds the original scheduling date (ODATE) by more than the number of days specified in the MAXWAIT parameter on the day the job is scheduled by RETRO=Y. For more information, see "RETRO: Basic Scheduling Parameter" on page 607.

Emergency jobs not belonging to a group are discarded if their specified MAXWAIT periods have passed. An emergency job that belongs to a specific group (specified in the GROUP parameter) and whose MAXWAIT period has not passed is not deleted from the Active Jobs file until all of the regular jobs that belong to the same group have finished executing. This is in case the job is needed at a later stage.

If the DUE OUT DAYS or TO DAYS parameter is specified, the job may stay in the Active Jobs file for the number of days equal to the larger value of days specified in these parameters.

For jobs containing a time zone later than the local CONTROL-M time, one day is added to MAXWAIT so that the job will stay one additional day on the Active Jobs file.

# **MAXWAIT Values for Jobs in a Group Scheduling Table**

The MAXWAIT value for jobs in a Group scheduling table is normally determined by the MAXWAIT parameter in the schedule tags defined in the Group entity. However:

- If the TAGMAXWT parameter in the CTMPARM member in the IOA PARM library is set to N (No), the MAXWAIT value for each job in the group is instead determined by the value of the MAXWAIT parameter in its job scheduling definition.
- If AND is specified in the RELATIONSHIP parameter, the MAXWAIT value from the job scheduling definition is used (regardless of the value of the TAGMAXWT parameter).
- If a job in a group is forced, the MAXWAIT value is determined by the value of the MAXWAIT parameter in the job scheduling definition, regardless of the value of the TAGMAXWT parameter.

## **MAXWAIT Values for Cyclic Jobs**

If a cyclic job is executing at the time the New Day procedure is run and the job's MAXWAIT value is reached, the New Day procedure changes the job to a non-cyclic job so that it can subsequently be deleted during the next New Day procedure.

# **Examples**

#### **Example 1**

If the original scheduling date of the job has passed, give the job an extra three days to be submitted.

Figure 233 MAXWAIT Parameter Example 1

```
JOB: OPERJOB LIB CTM.PROD.SCHEDULE
                                                                  TABLE: OPER
                                                                  SCROLL==> CRSR
  MEMNAME OPERJOB MEMLIB CTM.PROD.JCL
OWNER SYS1 TASKTYPE JOB PREVENT-NCT2 Y DFLT N
  APPL OPER
                                        GROUP MAINTENANCE
  DESC JOB RUN IN FIRST HALF OF THE MONTH
  OVERLIB
                                                              STAT CAL
  SCHENV
                                  SYSTEM ID
                                                              NJE NODE
  SET VAR
  CTB STEP AT NAME TYPE
DOCMEM OPERJOB DOCLIB CTM.PROD.DOC
  DAYS
        02,04,06
                                                                  DCAL
                                                                       AND/OR
  WDAYS
                                                                  WCAL
  MONTHS 1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- Y 11- Y 12- Y
  DATES
  CONFCAL
                   SHIFT
                                RETRO Y MAXWAIT 03 D-CAT
  MINIMUM
                   PDS
  DEFINITION ACTIVE FROM
                                  UNTIL
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
                                                                         11.17.00
```

Assume that the job does not run due to the absence of the required runtime resources. The job that is scheduled for the 2nd of the month can wait from the 2nd through the 5th to be executed.

On the 6th, the MAXWAIT period expires and the job scheduled for the 2nd is not executed. The jobs scheduled for the 4th and 6th wait for execution until the 7th and 9th.

## **Example 2**

The job can wait for execution indefinitely, until the runtime requirements for the job are satisfied:

MAXWAIT 99

#### **Example 3**

Schedule the job for every working day, even if the computer is not active. Give the job an extra day in which to be submitted.

Assume that calendar WORKDAYS, specified in the DCAL parameter, contains the values 15, 16, 18, and 19. The computer was offline from the 16th up to and including the 18th, and the 15th was the last date that the job was scheduled for execution.

Figure 234 MAXWAIT Parameter Example 3

```
JOB: PRDKPLO1 LIB CTM.PROD.SCHEDULE
                                                         TABLE: PRODKPL
COMMAND ===>
                                                         SCROLL===> CRSR
 MEMNAME PRDKPL01 MEMLIB CTM.PROD.JCL
 OWNER SYS1 TASKTYPE JOB PREVENT-NCT2 DFLT N
 APPL KPL
                                  GROUP PROD-KPL
       DAILY PRODUCTION - START OF APPL-PROD-KPL
 DESC
 OVERLIB
                                                      STAT CAL
                                                      NJE NODE
 SCHENV
                             SYSTEM ID
 SET VAR
 SET VAR
CTB STEP AT NAME
 DOCMEM PRDKPLO1 DOCLIB CTM.PROD.DOC
                                                         DCAL WORKDAYS
 DAYS
                                                              AND/OR
 WDAYS
                                                          WCAL
 MONTHS 1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- Y 11- Y 12- Y
 MINIMUM
                SHIFT
                          RETRO Y MAXWAIT 01 D-CAT
                PDS
 DEFINITION ACTIVE FROM
                              UNTIL
COMMANDS: EDIT, DOC, PLAN, JOBSTAT
                                                               11.17.00
```

Today is the 19th. The job is scheduled three times with the different original scheduling dates of the 16th, 18th and 19th.

The jobs on the 16th and 18th are disregarded on the 20th if they have not yet executed. The job on the 19th is disregarded only on the 21st.

#### **Example 4**

Schedule the job for every working day, even if the computer is not active. If it does not execute within the scheduled day, remove it from the Active Job file:

MAXWAIT 00

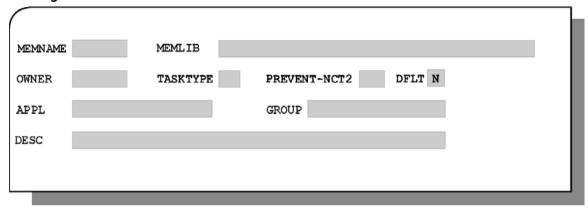
# **MEMLIB: General Job Parameter**

For a job (or warning messages): Name of the library containing the member specified in the MEMNAME parameter.

For a started task: Required started task information.

A related parameter is discussed in "OVERLIB: General Job Parameter" on page 571.

Figure 235 MEMLIB Parameter Format



Mandatory. Format of the parameter depends on whether the job scheduling definition applies to a job (or warning messages) or a started task:

■ For a job (or warning messages):

Valid values are a valid data set name of 1 through 44 characters, or one of the reserved values shown in Table 188.

**Table 188 MEMLIB Parameter Values for Non-Started Tasks** 

Value	Description
DUMMY	For dummy jobs.
	For warning messages, do not use DUMMY as a value for this parameter.
USER=name	For user-defined libraries.
GENERAL	Specifies the library referenced by the DALIB DD statement in the CONTROL-M procedure.
DDNAME=ddname	Specifies the library/member pointed to by the <i>ddname</i> DD statement in the CONTROL-M monitor procedure from which the JCL is to be submitted. Sample user exit CTMX002G in the IOA SAMPEXIT library must be installed.

#### ■ For a started task:

Any of the formats shown in Table 189 can be used for the MEMLIB value.

**Table 189 MEMLIB Parameter Formats for Started Tasks** 

Format	Description
*.taskid	Where <i>taskid</i> is the task ID. The STC is activated in the computer in which the CONTROL-M monitor is active.
cpuid, stcparms	Where <i>cpuid</i> is the ID of the computer in which the STC is to be activated (see the following value " <i>cpuid</i> "); <i>stcparms</i> represents STC parameters.
cpuid.taskid	Where <i>cpuid</i> is the ID of the computer in which the STC is to be activated (see the following value " <i>cpuid</i> "); <i>taskid</i> is the identifier to be specified on the start command.
cpuid	Where <i>cpuid</i> is the ID of the computer in which the STC is to be activated. Valid <i>cpuid</i> values are:
	<ul> <li>* – The same computer where the CONTROL-M monitor is active.</li> </ul>
	Under JES2:
	<ul> <li>Nn - Where n is the JES/NJE node ID.</li> <li>Mm - Where m is the machine ID.</li> <li>NnMm - Where n is the JES/NJE node ID, and m is the machine</li> </ul>
	ID.  When using the cpuid,stcparms format (see above), any quotes specified in the stcparms must be doubled.
	Under JES3
	■ Lname – Where name is the logical JES name of the machine, that is, the name as used in the JES3 command *T, not the SMF system ID.

## **General Information**

Whether the job scheduling definition applies to a job, warning messages, or a started task is determined by the values defined in the TASKTYPE parameter, which is described in "TASKTYPE: General Job Parameter" on page 654.

AutoEdit variables can be specified and are resolved. Even the machine ID, which is relevant for started task initiation, can be automatically replaced based on resource allocation. For more information, see Chapter 5, "JCL and AutoEdit Facility."

# For Jobs (or Warning Messages)

The library can be any standard cataloged partitioned data set (PDS or PDSE), LIBRARIAN or PANVALET. The record length must be 80.

The library and the member do not have to exist when the job production parameters are defined. Their existence is checked by CONTROL-M before actual submission of the job.

If, during the access to a library by CONTROL-M (before submission), the library is held exclusively by another user, such as a TSO user or job, the monitor tries to access the library every few seconds until the library is released and the job can be submitted. If the library is migrated, for example, through HSM, CONTROL-M remains in a WAIT state until the library is recalled.

Use of the library name DUMMY is intended for scheduling events, for example, adding a prerequisite condition without actually running the job. If the library name DUMMY is used, the job is not submitted; submission and sysout checking are skipped. In this case, the job is assumed to have ended OK (ON PGMST...DO processing is not performed), and Post-Processing parameters associated with an ENDED OK status are activated (OUT, SHOUT WHEN OK).

If the library name is GENERAL:

- The job is submitted from the library referenced by the DALIB DD statement of the CONTROL-M procedure. This library must be a partitioned data set or a concatenation of partitioned data sets.
- The standard ISPF Editor cannot process more than four concatenated libraries. The editor saves the edited member in the first library in the concatenation.
  - CONTROL-M Exit CTMX014 (the CTMX014G member in the IOA SAMPEXIT library) enables you to bypass these limitations if the members are going to be edited online through the J (JCL) option in the Job List screen or the Active Environment screen.
- The prefix USER= must be specified when a special type of user library is used. When using this prefix, the member is not read by CONTROL-M using the normal mechanism. Instead CONTROL-M submission Exit CTMX002 must be coded to handle access and submission of the library and member. In such cases, the CONTROL-M monitor ignores the data specified in the MEMLIB/MEMNAME parameters; however, the substring following the USER= may be used by the exit. For examples of the exit, see the IOA SAMPEXIT library.

When specifying option J (JCL) in the Job List screen or the Active Environment screen in order to edit the JCL member, CONTROL-M must determine which library (MEMLIB or OVERLIB) to use.

#### – NOTE –



The algorithm for this decision is described in "Editing A Member through The J (JCL) Option" on page 573.

#### For Started Tasks

A started task is activated in the specified computer ID. This is the ID of the computer in JES, not the 4-character SMF ID. You can use the \$D MEMBER JES command to determine the JES ID. For more information, see the discussion on specifying IOA CPUs in the description of the customization process in the *INCONTROL for z/OS Installation Guide*. If the computer ID is followed by a comma and parameters, the parameters are applied to the started task.

## **Examples**

#### **Example 1**

Submit the job from the IMSBKUP member in the SYS2.IMS.JOB library:

MEMNAME IMSBKUP
MEMLIB SYS2.IMS.JOB

#### **Example 2**

Activate started task COLCTSMF in the computer where the CONTROL-M monitor is operating:

MEMNAME COLCTSMF
MEMLIB *,DATE=%%ODATE

On September 5, the STC is activated by issuing the operator command:

S COLCTSMF, DATE=000905

#### **Example 3**

Activate started task GTF in the computer in which the CONTROL-M monitor is operating; task ID is G01:

MEMNAME GTF
MEMLIB *.G01

The STC is activated by issuing the operator command:

S GTF.G01

#### **Example 4**

Activate started task COLCTSMF on JES node 1:

MEMNAME COLCTSMF
MEMLIB N1,DATE=%%ODATE

#### **Example 5**

Activate started task COLCTSMF on machine 1 on JES node 1:

MEMNAME COLCTSMF
MEMLIB N1M1, DATE=%%ODATE

#### **Example 6**

Activate started task STAMSTC on MAS member 2 with identifier IDNTFIER:

MEMNAME STAMSTC MEMLIB M2.IDNTFIER

The STC is activated by issuing the operator command:

\$M2,'S STAMSTC.IDNTFIER'

# **MEMNAME: General Job Parameter**

Name of the member that contains one of the following (depending on the defined task type):

- JCL of the job
- Started task procedure
- Warning messages

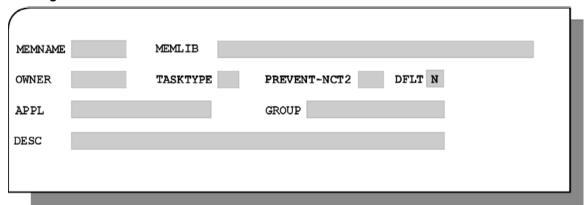
#### NOTE



For a Group Entity, this parameter has a different meaning, which is explained in "For a Group Entity" on page 529.

For more information, see "MEMLIB: General Job Parameter" on page 523.

Figure 236 MEMNAME Parameter Format



Mandatory. MEMNAME identifies a valid member name of 1 through 8 characters. For On Spool jobs, mask characters * and ? are supported. For details, see "Character Masking" on page 83 and "On Spool Jobs" on page 683.

#### NOTE -



CONTROL-M does not support members that have been compressed using the ISPF PACK option.

### **General Information**

The MEMNAME parameter identifies a member whose contents are determined by the task type of the job scheduling definition. For more information, see "TASKTYPE: General Job Parameter" on page 654.

- If TASKTYPE contains the value JOB, CYC, EMR or ECJ, the job scheduling definition is defined for a job and the MEMNAME parameter identifies the member that contains the JCL of the job.
- If TASKTYPE contains the value STC, CST, EST or ECS, the job scheduling definition is defined for a started task and the MEMNAME parameter identifies the member that contains the started task procedure.
- If TASKTYPE contains the value WRN, the job scheduling definition is defined for warning messages and the MEMNAME parameter identifies the member that contains the warning messages.

#### For a Job

The member name may be the same as or different than the job name.

The member can contain the JCL of more than one job. By default, CONTROL-M processes only the first job in the member. If, however, the MULTJOBS parameter in the CTMPARM member in the IOA PARM library is set to Y (Yes), CONTROL-M submits all the jobs in the member, but still only monitors the execution and results of the first job in the member. Therefore, BMC Software recommends that each member contain the JCL of only one job.

#### For a Group Entity

In a Group Entity, the MEMNAME parameter does not indicate a member name. Instead, MEMNAME is used for descriptive purposes in certain screens, such as in the NAME field of the Active Environment screen.

# **Example**

The JCL for job OPERCOMP is located in the OPERCOMP member in the library CTM.PROD.JCL.

Figure 237 MEMNAME Parameter Example

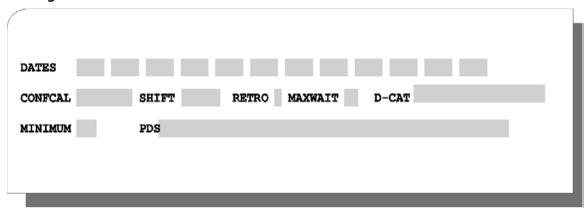
```
JOB: OPERCOMP LIB CTM.PROD.SCHEDULE
                                                      TABLE: OPER
COMMAND ===>
                                                      SCROLL===> CRSR
+----
 MEMNAME OPERCOMP MEMLIB CTM.PROD.JCL
 OWNER SYS1 TASKTYPE JOB PREVENT-NCT2 DFLT N
 APPL OPER GROUDESC JOB RUN ON THE 1ST OF THE MONTH
                                 GROUP MAINTENANCE
 OVERLIB
                                                   STAT CAL
  SCHENV
                            SYSTEM ID
                                                    NJE NODE
  SET VAR
 CTB STEP AT NAME TYPE
DOCMEM OPERCOMP DOCLIB CTM.PROD.DOC
 DAYS 01
                                                       DCAL
                                                           AND/OR
                                                       WCAL
 MONTHS 1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- Y 11- Y 12- Y
  DATES
  CONFCAL
               SHIFT
                             RETRO Y MAXWAIT OO D-CAT
 MINIMUM
                PDS
 DEFINITION ACTIVE FROM
                            UNTIL
 ΤN
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
                                                        11.17.00
```

# MINIMUM: Basic Scheduling Parameter

Minimum number of free tracks required by the library specified in the PDS parameter.

A related parameter is discussed in "PDS: Basic Scheduling Parameter" on page 579.

Figure 238 MINIMUM Parameter Format



Optional. However, if PDS is specified, MINIMUM is mandatory. The MINIMUM parameter specifies the minimum number of free tracks required. This must be a positive 3-digit number; leading zeros are inserted if necessary.

The MINIMUM parameter cannot be used with the DAYS, WDAYS, MONTHS, CONFCAL, RETRO and DATES parameters.

## **General Information**

The MINIMUM and PDS parameters are always used together and are never used with other Basic Scheduling parameters.

The PDS parameter identifies a library, and the MINIMUM parameter specifies the minimum number of free tracks required by that library.

These parameters are intended for use (that is, definition) in jobs and started tasks that compress, clean and/or enlarge libraries, or which issue a warning message to the IOA Log file (that is, if TASKTYPE=WRN) if the minimum number of free tracks is not available.

If the MINIMUM and PDS parameters are defined for a job, the scheduling of the job is not related to or dependent upon any date criteria. Instead, the job is scheduled if the actual number of free tracks available in the specified library is below the specified minimum at time of daily job ordering. The job or started task can then compress, clean, or enlarge the library (or issue the appropriate warning).

#### - NOTE



MINIMUM does not work with PDSE-type libraries because they always appear to be 100 percent full. MINIMUM only checks current extents.

# **Examples**

#### **Example 1**

Schedule the job when there are less than 20 unused tracks in the library ALL.PARMLIB.

Figure 239 MINIMUM Parameter – Example 1

```
JOB: OPERCOMP LIB CTM.PROD.SCHEDULE
                                                             TABLE: OPER
COMMAND ===>
                                                             SCROLL===> CRSR
 MEMNAME OPERCOMP MEMLIB GENERAL
OWNER SYS1 TASKTYPE JOB PREVENT-NCT2 DFLT N
 APPL OPER DESC COMPR
                                    GROUP OPER-MAINT
         COMPRESS OF ALL.PARMLIB
  DESC
  OVERLIB
                                                          STAT CAL
  SCHENV
                               SYSTEM ID
                                                          NJE NODE
 SEI VAR
CTB STEP AT NAME
  SET VAR
                                    TYPF
 DOCMEM OPERCOMP DOCLIB CTM.PROD.DOC
 DAYS
                                                              DCAL
                                                                   AND/OR
  WDAYS
                                                              WCAL
  MONTHS 1- 2- 3- 4- 5- 6- 7- 8- 9- 10- 11- 12-
 DATES
 CONFCAL SHIFT RETRO Y MAXWAIT 00 D-CAT MINIMUM 020 PDS ALL.PARMLIB
  DEFINITION ACTIVE FROM UNTIL
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
                                                                    11.17.00
```

#### **Example 2**

Send a warning message when there are less than 50 unused tracks in the library USER.LIBRARY:

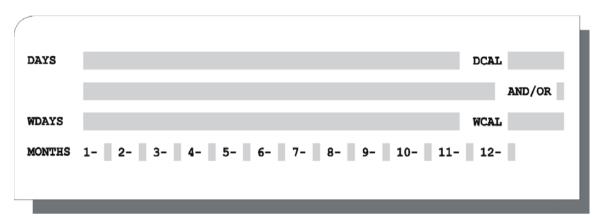
Figure 240 MINIMUM Parameter – Example 2

```
MEMNAME MSG001
TASKTYPE WRN
PDS USER.LIBRARY
MINIMUM 050
```

# **MONTHS: Basic Scheduling Parameter**

Months of the year in which the job must be scheduled.

Figure 241 MONTHS Parameter Format



Optional. The months in the year are represented by the numbers 1 through 12. A value can be specified for each month. Valid values are:

Table 190 MONTHS Parameter Values

Value	Description
Y (Yes)	Schedule the job in that month. Default.
N (No) or blank	Do not schedule the job in that month.

### **General Information**

In general, the job is scheduled for execution only during the months in which a value of Y is specified and only if the job would have also been scheduled due to some other basic scheduling parameter (such as DAYS, WDAYS, and so on); that is, the MONTHS parameter serves as a limiting filter. There are certain exceptions that are noted below.

The MONTHS parameter cannot be used with the PDS and MINIMUM parameters.

If values are set for both the MONTHS parameter and the DATES parameter, the MONTHS parameter setting is ignored.

A job can be scheduled in a month not specified as a working month if a greater than or less than qualifier in the DAYS specification shifts the scheduling out of the current month, and the month to which it shifts is a non-scheduled month, the job is nevertheless scheduled in that non-scheduled month.

#### **Example**

If the values of the DAYS parameter >31, the MONTHS parameter indicates JANUARY and MARCH (but not FEBRUARY).

The associated calendar has all days except JANUARY 31 as working days.

Then the job is scheduled on February 1.

# **Examples**

#### **Example 1**

Schedule a job only in March and September:

```
MONTHS 1- N 2- N 3- Y 4- N 5- N 6- N 7- N 8- N 9- Y 10- N 11- N 12- N
```

#### **Example 2**

Schedule job OPERCOMP on the first day of every month.

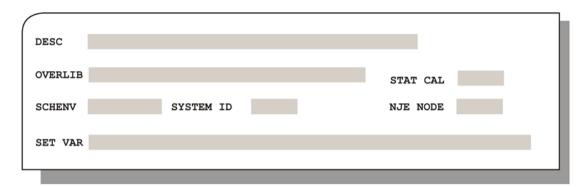
Figure 242 MONTHS Parameter – Example 2

```
JOB: OPERCOMP LIB CTM.PROD.SCHEDULE
                                                               TABLE: OPER
COMMAND ===>
                                                               SCROLL===> CRSR
 MEMNAME OPERCOMP MEMLIB CTM.PROD.JCL
 OWNER SYS1 TASKTYPE JOB PREVENT-NCT2 DFLT N
 APPL OPER GROUDESC JOB RUN ON THE 1ST OF THE MONTH
                                     GROUP MAINTENANCE
  OVERLIB
                                                          STAT CAL
 SCHENV
                               SYSTEM ID
                                                          NJE NODE
 SET VAR
 CTB STEP AT
 CTB STEP AT NAME TYPE DOCMEM OPERCOMP DOCLIB CTM.PROD.DOC
  DAYS
                                                               DCAL
                                                                   AND/OR
                                                               WCAL
  MONTHS 1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- Y 11- Y 12- Y
  DATES
                  SHIFT
                              RETRO Y MAXWAIT OO D-CAT
  CONFCAL
  MINIMIM
                  PDS
 DEFINITION ACTIVE FROM
                                UNTIL
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
```

# **NJE NODE: General Job Parameter**

Identifies the node in the JES network at which the job is to execute.

Figure 243 NJE NODE Parameter Format



NJE NODE specifies a node name of 1 through 8 characters. Only trailing blanks are allowed.

By default, the NJE NODE parameter is optional.

## **General Information**

The NJE NODE parameter is used to identify the node in the JES network at which the job is to execute.

If a value is specified for the NJE NODE parameter, a JCL statement is generated. The precise form of the statement depends on whether CONTROL-M is running under JES2 or JES3.

If the task type is a started task, NJE NODE is protected. If the task type is changed from a job to a started task, NJE NODE is erased and protected.

## **Under JES2**

If CONTROL-M is running under JES2, the NJE NODE parameter generates the following JCL statement:

/*ROUTE XEQ node_name

### **Under JES3**

If CONTROL-M is running under JES3, the JCL statement generated by the NJE NODE parameter differs slightly, taking the following form:

//*ROUTE XEQ node_name

Note that if a JES3 NJB job statement is not present in the job, the //*ROUTE XEQ JCL statement is not generated.

If a value is specified for the NJE NODE parameter, it will not override any node name specified in the job statement unless the OVERJCLM parameter in the CTMPARM library is set to Y.

## **Examples**

#### **Example 1**

CONTROL-M is running under JES2. The following is specified:

DESC		
OVERLIB		STAT CAL
SCHENV	SYSTEM ID	NJE NODE OS35

The following statement is added to the JCL of the job:

/*ROUTE XEQ OS35

and the job is executed at node OS35.

#### **Example 2**

CONTROL-M is running under JES3. The following is specified:

DESC		
OVERLIB		STAT CAL
SCHENV	SYSTEM ID	NJE NODE OS35

The following statement is added to the JCL of the job:

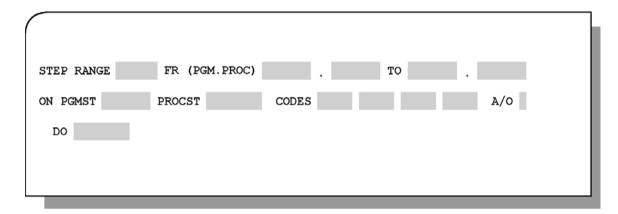
//*ROUTE XEQ OS35

and the job is executed at node OS35.

# **ON Statements: Post-Processing Parameter**

Job processing criteria that determine whether the accompanying DO statements are performed.

Figure 244 ON Statement Format Example



ON statements identify specific steps in the execution of a CONTROL-M job or group.

The types of ON statement are described in Table 191. Each type is discussed in detail in this chapter.

**Table 191 ON Statement types** 

ON Statement Type	Description
ON GROUP-END	Group processing criteria that determine whether the accompanying DO statements are performed For more information, see "ON GROUP-END: Post–Processing Parameter" on page 539.
ON PGMST	Job processing step and code event criteria that determine whether the accompanying DO statements are performed For more information, see "ON PGMST: Post–Processing Parameter" on page 542.
ON SYSOUT	Search for a string in the sysout of the job, and perform the accompanying DO statements if the string is found For more information, see "ON SYSOUT: Post–Processing Parameter" on page 559.

Multiple ON statements can be specified.

Multiple ON PGMST and ON SYSOUT statements can be linked by Boolean logic.

## **General Information**

ON statements are usually, but not necessarily, followed by user-specified DO actions. The implied relationship between ON statements and associated DO statements is: if ON statement criteria are satisfied, perform the associated DO statement actions.

The combination of ON statements and DO statements enables you to specify post-processing actions that depend on the execution results of job steps executed under CONTROL-M.

### **Multiple ON Statements and ON Blocks**

In a new job scheduling definition, an empty ON statement is followed by an empty DO statement. Additional ON statements can be opened in the job scheduling definition as required.

For more information, see the topic, "Multiple ON Statements and ON Blocks," under the relevant ON statement type.

# **ON GROUP-END: Post-Processing Parameter**

Group processing criteria that determine whether the accompanying DO statements are performed. Found in Group Entities only.

Figure 245 ON GROUP-END Parameter Format



Optional. Valid values are shown in Table 192.

Table 192 ON GROUP-END Values

Value	Description
OK	Process the accompanying DO statements if all scheduled jobs in the group ended OK.
NOTOK	Process the accompanying DO statements if not every job in the group ended OK.

### **General Information**

The ON GROUP-END parameter enables specification of DO statements to be performed when the processing of the group ends with the indicated status.

By default, if not all jobs in the group ended OK, the DO statements accompanying an ON GROUP-END NOTOK parameter are performed. This applies if at least one job ended NOTOK, and it can also apply if a job in the group was deleted and all remaining jobs in the group ended OK. However, if the GRPDELJB parameter in the CTMPARM member in the IOA PARM library is set to Y (Yes), deleted jobs are not considered, and status END NOTOK applies only if at least one job ended NOTOK.

If the job that ended NOTOK is subsequently successfully rerun, so that the termination status of the group changes to OK, the DO statements accompanying an ON GROUP-END OK parameter are then performed.

The following DO statements can be specified following an ON GROUP-END statement:

- DO COND
- DO FORCEJOB
- DO NOTOK
- DO OK
- DO SET
- DO SHOUT
- DO MAIL

DO OK or DO NOTOK statements change the final status of the group, not the status of each job or job step in the table.

Use of the ON GROUP-END parameter in the Group Entity can frequently reduce the number of individual DO statements that would otherwise require definition in individual job scheduling definitions.

For example, suppose that following the processing of the group, you want to force a particular job if any of the jobs in the group ENDED NOTOK.

- This result can be achieved by defining an ON GROUP-END NOTOK DO statement (in the Group Entity) followed by the appropriate DO FORCEJOB statement.
- To achieve this result without use of the ON GROUP-END parameter, the following steps would be necessary:
  - In each job scheduling definition in the table, define an appropriate condition that would be added to the IOA Conditions file when the job ends NOTOK.
  - In the table, define an additional job to be performed after the other jobs in the table have terminated. This job would have as an IN condition the condition added by the jobs that ended NOTOK, and would trigger the appropriate job.

### Multiple ON GROUP END Statements and ON GROUP END Blocks

Multiple ON GROUP-END parameters can be defined. Upon specifying an ON GROUP-END value and pressing **Enter**, a new ON GROUP-END statement, followed by a blank DO statement, is opened.

# **Example**

If a job in the Group scheduling table ACCOUNTS_GROUP ended NOTOK, add condition ACCTS-CHK-REQUIRED.

Figure 246 ON GROUP-END Parameter Example

```
GRP ACCOUNTS GROUP CTM.PROD.SCHEDULE(GRP)
COMMAND ===>
                                                           SCROLL===> CRSR
  SCHEDULE TAG
  DAYS
                                                            DCAL
                                                                AND/OR
                                                           WCAL
  MONTHS 1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- Y 11- Y 12- Y
  DATES
  CONFCAL
            SHIFT
                           RETRO N MAXWAIT 00
  SCHEDULE TAG ACTIVE FROM
                            UNTIL
  ΙN
  CONTROL
  FROM TIME + DAYS UNTIL TIME + DAYS
DUE OUT TIME + DAYS PRIORITY SAC CONFIRM
  TIME ZONE:
  OUT
  ON GROUP-END NOTOK
   DO COND ACCTS-CHK-REQUIRED ODAT +
  SHOUT WHEN
                    TIME + DAYS TO
                                                                URGN
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
                                                                  18.19.14
```

# **Confirmation panel**

If the DFJCONF profile variable is set to Y, and the JOB parameter in the DO FORCEJOB request is blank, a confirmation panel is displayed when exiting the Job Scheduling Definition screen. The confirmation panel is displayed only once for each DO FORCEJOB statement.

Figure 247 ON GROUP-END Confirmation Panel

```
THIS JOB CONTAINS ONE OR MORE DO FORCEJOB STATEMENTS.

WHEN THE JOB IS ORDERED:

ARE YOU SURE YOU WANT TO FORCE THE WHOLE TABLE IN THE

FORCEJOB STATEMENT(S)? (Y/N)
```

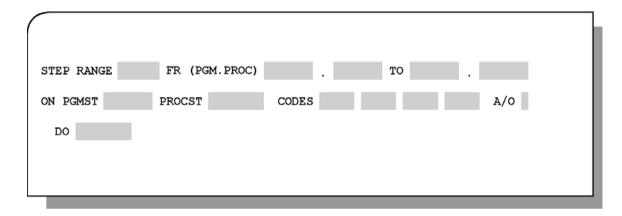
Enter Y to save the scheduling table and return to the Job List screen, or N to return to the scheduling table without saving it.

# **ON PGMST: Post-Processing Parameter**

Job processing step and code event criteria that determine whether the accompanying DO statements are performed.

For more information, see "STEP RANGE: Post-Processing Parameter" on page 641.

Figure 248 ON Parameter Format



Optional. ON PGMST statements define event criteria that identify specific CONTROL-M job steps and possible codes that result from the execution of those job steps.

The ON PGMST statement consists of the subparameters described in Table 193. When used, at least one step and one code must be specified.

**Table 193 ON PGMST Parameter Subparameters (part 1 of 2)** 

Subparameter	Description
PGMST	Job step. The execution results of the program executed by the job step are checked against the specified CODES criteria. From 1 through 8 characters. Mandatory. Valid values are:
	■ pgmstep – Name of the step (EXEC statement): //pgmstep EXEC PGM=program The ON PGMST statement is satisfied only when the program execution results from the specified step satisfy the specified code criteria. For more information, see "PGMST" on page 547.
	*rangename – Range name. rangename is the name of a step range defined in the STEP RANGE parameter. The asterisk (*) preceding the name indicates to CONTROL-M that the specified name is a range name, not a step name. For more information, see "STEP RANGE" on page 547, and "STEP RANGE: Post–Processing Parameter" on page 641.
	<b>Note</b> : Some third party products produce JCL step names that begin with an * (asterisk) character. If you specify a JCL step name of this type in an ON PGMST statement, CONTROL-M interprets this step name as a step range.
	The solution is to define a workaround step range that includes only the problematic step name.
	For example, to process the step name *OMVTEX, use the following: STEP RANGE ONESTEP FR (PRM.PROC) *OMVTEX . TO *OMVTEX ON PGMST *ONESTEP PROCST CODES xxxxx
	■ ANYSTEP – any job step Generally, the ON PGMST statement is satisfied when the program execution results from any job step satisfy the specified code criteria. For more information, including the exceptions, see "Step Name: ANYSTEP" on page 548.
	■ +EVERY – every job step The ON PGMST statement is satisfied if the program execution results from every job step that satisfies the specified code criteria. For more information, see "Step Name: +EVERY" on page 548.
	■ \$FIRST – the first-executed job step
	■ \$LAST – the last-executed job step
	<b>Note</b> : The \$FIRST and \$LAST values have these special meanings only if the LASTSTEP parameter in the CTMPARM member is set to Y.
	Neither CONTROL-M/Restart steps nor FLUSHed steps are considered the first or last step for this purpose.

**Table 193 ON PGMST Parameter Subparameters (part 2 of 2)** 

Subparameter	Description		
PROCST	Procedure step (EXEC statement) that invokes a procedure from which the specified PGMST program is executed. 1 to 8 characters. Optional. Valid values are: <ul> <li>" " (blank) - When the PROCST field is left blank, matching program step names (PGMST) are checked regardless of whether they are directly from the job or from a called procedure.</li> </ul>		
	Default. The ON PGMST statement is satisfied if the PGMST criteria are satisfied from any procedure directly from the job.		
	<ul> <li>procstep – Name of a specific procedure step:</li></ul>		
	<ul> <li>+EVERY         Matching program step names (PGMST) are checked from all called procedures and from the job itself.         The ON PGMST statement is satisfied only when the code criteria for the program step are satisfied for all occurrences (called procedures and directly in the job stream). For more information, see "Step Name: +EVERY" on page 548.     </li> </ul>		
CODES	Return codes or statuses that can satisfy the step or code event criteria if returned upon termination of the specified job steps. At least one code must be specified. CODES can be condition codes, user abend codes, system abend codes, various end codes and statuses, and certain keywords. CODES are discussed in "General Information," immediately below this table.		
A/O	Optional. Specifying either A (And) or O (Or) opens a new ON PGMST statement in the ON PGMST block (described in "Multiple ON PGMST Statements and ON PGMST Blocks" below this table) and links the new statement to the statement containing the A/O specification, as follows:		
	<ul> <li>A (And) – Indicates AND logic between the two ON PGMST statements. ON PGMST block criteria are satisfied only if both ON PGMST statements are satisfied.</li> </ul>		
	<ul> <li>O (Or) – Indicates OR logic between the two ON statements. ON PGMST block criteria are satisfied if either (or both) ON PGMST statements are satisfied.</li> </ul>		

## **General Information**

ON PGMST statements are usually, but not necessarily, followed by user-specified DO actions. The implied relationship between ON PGMST statements and associated DO statements is described in "General Information" on page 538.

### Multiple ON PGMST Statements and ON PGMST Blocks

In a new job scheduling definition, an empty ON PGMST statement is followed by an empty DO statement. Additional ON PGMST statements can be opened in the job scheduling definition as follows:

■ When you set values for ON PGMST, PROCST, and CODE, and press Enter, an empty ON PGMST and DO statement is opened following the current ON PGMST and DO statements. The new ON PGMST and DO statements, if filled in, are not logically connected to the preceding ON PGMST and DO statements. They constitute a new ON PGMST block and DO block.

Multiple ON PGMST blocks are normally interpreted sequentially. If the conditions of an ON PGMST block are satisfied, the accompanying DO actions are performed. The conditions of more than one ON PGMST block can be satisfied; therefore, more than one set of DO statements can be performed.

### **Example**

One ON PGMST block specifies STEP1 as the program step, and >C0004 as the CODE.

A second ON PGMST block specifies ANYSTEP as the program step, and >C0008 as the CODE.

If STEP1 results in a condition code of C0016, the ON PGMST step and CODE event criteria for both ON PGMST statements are satisfied, and the DO actions accompanying both ON PGMST blocks are performed.

■ When you fill in the A/O (And/Or) subparameter of an ON PGMST statement, an empty ON PGMST statement is opened immediately, that is, before the accompanying DO statement. The specified A/O value logically connects the new ON PGMST statement to the preceding ON PGMST statement. These two ON PGMST statements constitute a single ON PGMST block.

### Example

```
ON PGMST STEP1 ... CODES CO004 ... A/O A
ON PGMST STEP5 ... CODES SOC4 ... A/O
DO SHOUT ...
```

In the above ON PGMST and DO statements, for the DO SHOUT action to be performed, STEP1 must end with a condition code of C0004, and STEP5 must end with an S0C4 system abend.

To add an empty ON PGMST statement between two existing ON PGMST statements, type the > character over the first letter in the ON PGMST value of the previous ON PGMST line, and press **Enter**.

### **Example**

If the program step name is STEP1, type the following and press **Enter**:

ON PGMST >TEP1

This adds an "empty" ON PGMST line after the current ON PGMST statement. The STEP1 step name is restored to its original value when **Enter** is pressed (that is, the > character disappears and the S character is restored).

To delete unwanted ON PGMST statements, specify appropriate Line Editing commands in the Edit environment. For information on the Edit environment, see Appendix C, "Editing Job Scheduling Definitions in the Edit Environment," and in particular "Line Editing Commands" on page 915.

# §Restart§ Using All Runs of a Job Including Restarts

When processing ON PGMST blocks, CONTROL-M can incorporate the results of all previous runs and restarts, filtering them for jobs restarted with the RESTART, RECAPTURE CONDITION, and ABEND CODES parameters. CONTROL-M/Restart searches previous runs to determine which steps must be considered part of the restarted job.

For example, if one step finished successfully during its original run and another step finished successfully after a restart, the ON block check for the successful finish for both steps produces a TRUE result and the ON statement is satisfied.

Activation of this facility requires that the ALLRUNS parameter in the CTRPARM member be set to YES. When activated, this facility can apply to any specified step, step range, or to the ON PGMST step value +EVERY. **\$Restart\$** 

#### NOTE



Post-processing of ON PGMST statements during a RESTART or RERUN is independent of the post-processing of the same ON PGMST statements during the earlier run. In these situations, you may get duplicate actions.

## **Step Values**

### **PGMST**

Within an ON PGMST statement, the specified step is generally a program step, specified in the PGMST field. It may be a program executed directly within the job stream, in which case no PROCST value is specified, or it may be a program executed by a called procedure, in which case the called procedure is specified in PROCST.

If the JCL contains nested procedures, the name of the EXEC procedure statement that invokes the most deeply nested procedure, that is, the procedure that immediately invokes the PGM step, must be specified in PROCST.

The same step name can appear in different ON PGMST statements in the same ON PGMST block, or in different ON PGMST blocks.

### **STEP RANGE**

To check codes in a range of steps, first define the step range and assign it a name in the STEP RANGE statement, which is described in "STEP RANGE: Post–Processing Parameter" on page 641. Then specify the name, preceded by an asterisk, in the PGMST field. The * indicates that the specified name is a range name, not a step name. The range of steps is displayed, and you can check the codes that are displayed within the defined range.

If the LASTSTEP parameter is set to Y in the CTMPARM member, CONTROL-M treats the job step named \$LAST as the job step executed last, and the job step named \$FIRST as the job step executed first.

If CONTROL-M adds a CONTROL-M/Restart step to a job, for example, if a job is restarted by CONTROL-M/Restart, or if PREVENT NCT2 is specified in the job scheduling definition, the CONTROL-M/Restart step is processed like all other job steps.

CONTROL-M does not treat the CONTROL-M/Restart step or any FLUSHED step as the step executed last or first.

### **Example 1**

In the STEP RANGE statement, name DF2 is assigned to the range of program steps STEP20 through STEP29A.

If *DF2 is specified in ON PGMST, the ON step and code criteria is satisfied if any of the codes result from any of the steps in the range STEP20 through STEP29A.

## Example 2

You want to define a job in such a way that its status depends on the result of the last step executed. If the last step ended with a condition code of 0, give the job the status ENDED OK. If the last step ended with any other condition code, the job status is to be ENDED NOTOK.

The following are sample ON PGMST statements for such a job:

ON PGMST ANYSTEP	PROCST	CODES C****	U**** S***	A/0
DO OK				
ON PGMST \$LAST	PROCST	CODES >COOOO	U**** S***	A/0
DO NOTOK				

# **Step Name: ANYSTEP**

You can specify ANYSTEP as the value in the PGMST field. In general, it indicates that the DO statements must be performed if the specified codes are found in any steps.

However, if ANYSTEP is specified with codes OK, NOTOK, EXERR, JLOST, JNRUN, JSECU, JNSUB or *UKNW, the ON criteria are satisfied only if the entire job ends with the specified code criteria.

If ANYSTEP is specified with code FORCE, no other codes can be specified in the same ON block, and the PROCST parameter must be left blank. For a description of code FORCE, see "Valid CODES Values" on page 551.

# Step Name: +EVERY

The value +EVERY is used without being accompanied by limiting step values when the code criteria must be satisfied for every step. The following examples all have the same impact – the code criteria must be satisfied for every step in the job without exception.

### NOTE



A DO OK or DO NOTOK statement is ignored if it is specified in an ON PGMST +EVERY statement.

**Examples** 

- ON PGMST +EVERY PROCST
- ON PGMST ANYSTEP PROCST +EVERY

The ANYSTEP value is not a limiting value. In this case, it has the same meaning as +EVERY.

■ ON PGMST +EVERY PROCST +EVERY

Value +EVERY is generally accompanied by a limiting step value when the code criteria must be satisfied for every step within the specified limits, as follows:

■ If the limiting value is a PROCST value, the code criteria must be satisfied by all job steps from within the specified procedure.

### **Example**

```
- ON PGMST +EVERY PROCST STEP1
```

Every program step of procedure step STEP1 must be satisfied.

■ If the limiting value is a PGMST value, the code criteria must be satisfied by all executions of the specified job step (or range of steps if a range is specified), from within the job steam and within all procedures.

### **Examples**

```
- ON PGMST StepA PROCST +EVERY
```

All executions of job step STEPA from within the job stream and within every procedure must be satisfied.

```
— ON PGMST *Range1 PROCST +EVERY
```

Executions of all job steps in Range1, from within the job stream and within every procedure, must be satisfied.

Step name +EVERY can be specified with the following codes: Cnnnn, Sxxx, Unnnn, *xxxx, FLUSH, SNRUN and *****.

- When step name +EVERY is specified with codes *Cnnnn*, *Sxxx*, *Unnnn* and **xxxx*, the following conditions must be satisfied to satisfy the ON statement:
  - If the steps that run (excluding FLUSH steps) satisfy the PGMST and PROCST criteria, they must also not contradict the Cnnnn, Sxxx, Unnnn or *xxxx codes.
  - At least one step runs and fulfills the above conditions.
- When step name +EVERY is specified with codes FLUSH, SNRUN or *****, the following apply:
  - ON PGMST +EVERY CODES FLUSH is satisfied if in each job step, a JCL COND or JCL IF/THEN/ELSE statement caused the step not to run.
  - ON PGMST +EVERY CODES SNRUN is satisfied if each job step did not run.
  - ON PGMST +EVERY CODES ***** is satisfied if each defined job step ran and no job step was flushed (that is, due to a JCL COND or JCL IF/THEN/ELSE statement).

### **CODES Values**

CODES can be condition codes, user abend codes, system abend codes, various end codes and statuses, and certain keywords. They can also be prefaced by certain qualifiers. All of these are described below.

A maximum of 245 values can be specified for CODES in any ON PGMST statement, as follows:

- Each line of an ON PGMST statement contains fields for specification of up to four values for CODES.
- Whenever a fourth value is specified on a line for CODES, and Enter is pressed, a new line within the same ON PGMST statement is opened, allowing specification of as many as four additional CODES values.

## **Valid CODES Values**

## — NOTE -



A DO OK statement specified in the job scheduling definitions is ignored if:

- any of the following status codes apply to the job:
  - EXERR
  - JNSUB
  - *REC0
  - *UKNW

-or-

■ the DO OK statement was specified as part of an ON PGMSTEP ANYSTEP *pgmstep* CODE NOTOK condition, because if that condition is satisfied, the status of the job has already been set to NOTOK

## Table 194 ON PGMST Parameter CODES Values (part 1 of 3)

Value	Description		
\$EJ	Job was queued for re-execution.		
****	Any step that executes (including steps with JCL errors and steps returned with an ABEND code). For reasons of backward compatibility, the CODES value ***** does not include steps with code FLUSH or SNRUN (described below). The CODES value ***** does, however, include jobs not submitted and jobs whose sysout was lost if ON PGMST ANYSTEP is specified.		
	<b>Note:</b> Although the CODES value **** includes steps which have returned any system abend code, the preferred method of indicating these steps is S***.		
*NCT2	A NOT CATLGD 2 or NOT RECATLGD 2 event occurred in the job step. The default result of this event is a NOTOK status for the step. A message containing the data set name is written to the IOA Log file.		
	Note: If you do not want to be alerted to NOT RECATLGD 2 events, see your INCONTROL administrator.		
*REC0	Rerun (recovery) is needed, but no more reruns are available.		
	<b>Note</b> : This status code is REC followed by a zero (not the letter O).		
*TERM	Job terminated by CMEM due to an NCT2 event.		
*UKNW	An unknown error occurred, usually as a result of a computer crash during job execution. This value can only be specified with step value ANYSTEP.		
*XXXX	Any step completion code (condition, system abend, user abend) that matches the string, where <i>x</i> can be any hexadecimal character (0 through 9, A through F) in user-defined events, which are turned on by Exit 3. Regarding usage, see your INCONTROL administrator.		
Cnnnn	Step condition code, where <i>nnnn</i> is a 4-digit value.		

Table 194 ON PGMST Parameter CODES Values (part 2 of 3)

Value	Description		
EXERR	Any type of execution error. It is the same as NOTOK, but is triggered only if the job has actually started executing. This value can only be specified with step value ANYSTEP.		
FLUSH	A JCL COND or JCL IF/THEN/ELSE statement caused a step to no run. This CODES value is described in more detail below.		
FORCE	This code applies when a job is FORCEd OK from the Active Environment screen (Screen 3). To specify a code of FORCE, all of the following must apply:		
	<ul> <li>No other code can be specified in the same statement.</li> <li>The PGMST value must be ANYSTEP.</li> <li>No PROCST value can be specified.</li> <li>No other ON statements can appear in the ON PGMST block.</li> </ul>		
	Valid DO statements for the FORCE code are:		
	<ul> <li>■ DO SHOUT</li> <li>■ DO COND</li> <li>■ DO FORCEJOB</li> <li>■ DO SETVAR</li> <li>■ DO MAIL</li> </ul>		
JFAIL	Job failed due to JCL error.		
JLOST	Job sysout was lost. This value can be specified only with step value ANYSTEP.		
JNRUN	Job was canceled during execution or re-execution. This value can be specified only with step value ANYSTEP.		
JNSUB	Job not submitted. Submission of a job or initiation of a started task failed for any reason. This value can be specified only with step value ANYSTEP.		
JSECU	Job failed due to security requirements (only under ACF2). This value can be specified only with step value ANYSTEP.		
NOTOK	A status of execution of the whole job.		
	This CODES value can only be specified with step value ANYSTEP. It indicates that at least one PGM step, or the whole job, finished executing NOTOK, meaning, with a condition code greater than that set as the upper limit. By default, this limiting condition code is C0004, but the MAXCCOK parameter in the CTMPARM member in the IOA PARM library can be used to set the default condition code to another value, such as C0000.		
	This CODES value covers all types of failures, including non-execution errors such as job not run, JCL error, or job not submitted.		

Table 194 ON PGMST Parameter CODES Values (part 3 of 3)

Value	Description
OK	A status of execution of the whole job.  This CODES value can only be specified with step value ANYSTEP.
	It indicates that all non-flushed PGM steps finished executing OK, meaning, with a condition code equal to or less than the condition code set as the upper limit. By default, this limiting condition code is C0004, but the MAXCCOK parameter in the CTMPARM member in the IOA PARM library can be used to set the default condition code to another value, such as C0000.
	If a job is FORCEd OK, the DO statements following an ON PGMST ANYSTEP <i>pgmstep</i> CODES OK statement are processed only if the FRCOKOPT parameter in the CTMPARM member in the IOA PARM library is set to Y (Yes).
SNRUN	A step did not run. This CODES value is described in more detail below.
Sxxx	Step system abend code, where xxx is a 3-character hex value.
Unnnn	Step user abend code, where <i>nnnn</i> is a 4-digit value.

## **FLUSH**

The CODES value FLUSH generally applies when a step does not run but no error is indicated. This CODES value is assigned in the following cases:

- A JCL COND or JCL IF/THEN/ELSE statement caused the step not to run.
   CONTROL-M detects CODES value FLUSH steps by message IEF272I (Step was not executed).
- **§Restart§** If a job was restarted by CONTROL-M/Restart, and CONTROL-M is to consider all job runs during post-processing (the ALLRUNS parameter is set to YES in the CTRPARM member), a step is defined as FLUSH if:
  - either the step did not previously run, or CONTROL-M/Restart did not recapture a completion or abend code from a previous run

recuprate a completion of abena code from a previous run

— it was not executed during the RESTART run because of a JCL COND or JCL IF/THEN/ELSE statement

or

and either

— it was not executed due to a RESTART decision (message CTR103I)

Because a CODES value of FLUSH does not indicate that an error occurred during job execution, assignment of this status does not cause a job status of NOTOK.

If a JCL statement other than the COND or IF/THEN/ELSE statement caused the step not to run, it is not defined as a FLUSH step.

If the failure of a step causes subsequent steps not to be executed, these subsequent steps are not defined as FLUSH steps.

For reasons of backward compatibility, that is, to ensure that the application of CODES value ***** remains unchanged, CODES value ***** does not include FLUSH steps.

## **SNRUN**

A step is defined as CODES value SNRUN if it did not run. This code includes

- any step with a CODES value of FLUSH
- any step that does not appear in the job
- instances where a step does not run because of a JCL error in a prior step (the step with the JCL error does not have a status of SNRUN)
- **§Restart§** if a job was restarted by CONTROL-M/Restart, and CONTROL-M is to consider all job runs during post-processing (the ALLRUNS parameter is set to YES in the CTRPARM member), a step is defined as SNRUN if one of the following conditions are satisfied:
  - The step did not previously run, or CONTROL-M/Restart did not recapture a completion or abend code from a previous run.
  - The step was not executed during the RESTART run.

SNRUN cannot be specified together with ANYSTEP. Because SNRUN includes steps that do not exist in a job, and ANYSTEP includes all step names even if they do not exist in a job, specifying both in the same job causes a condition that SNRUN cannot process.

A status of SNRUN does not indicate that an error occurred during a job execution, nor does it cause a job status of NOTOK. It merely indicates that it did not run.

For reasons of backward compatibility, that is, to ensure that the application of CODES value ***** remains unchanged, CODES value ***** does not include SNRUN steps.

# **Code Qualifiers and Relationships**

Any character in a condition code, system abend code or user abend code may be replaced by an asterisk (*). An asterisk means "any value" for the character it replaces. For example, if S*13 is specified, the code criteria for the step is satisfied by codes S013. S613. S913. and so on.

The qualifiers described in Table 195 can be used in certain cases.

### **Table 195 ON PGMST Parameter Code Qualifiers**

Qualifier	Description
>	Greater than. Valid as a qualifier for condition codes and user abend codes.
<	Less than. Valid as a qualifier for condition codes and user abend codes.
N	Specifies not to perform the accompanying DO statements if the specified code exists in the step. Valid as a qualifier for condition codes, user abend codes and system abend codes.

### NOTE



The N qualifier indicates that the DO statements must not be performed if the specified condition exists. It does not indicate that the DO statements must be performed if the specified condition does not exist.

The relationship between multiple codes in an ON PGMST statement is OR, that is, the appearance of any of the codes in the specified step satisfies the ON criteria, except for range specifications such as >C0010 or <C0040.

However, code criteria qualified by N take precedence over all other code criteria. If a code that is specified with an N qualifier is generated by the specified step, accompanying DO actions are not performed even if other ON code criteria are satisfied.

# **Examples**

- If >C0008 NC0020 is specified, the codes criteria is satisfied (and the DO statements performed) by the appearance of any condition code greater than 8 except condition code 20.
- If the following are specified:

>U0999 NU1341 S*** NS*37 <C0004

The DO actions are triggered by one of the following:

- a condition code less than C0004
- a user abend code greater than U0999 except U1341
- any system abend code except Sx37 (that is, except S037, S137, and so on)
- If only code NC0008 is specified:

The accompanying DO statements are never performed. The specified value only indicates when not to perform the DO actions. There is no indication when the DO actions are to be performed.

## **Example 1**

Any program step resulting in condition code C0008 or C0016 is considered OK.

Figure 249 ON PGMST Parameter – Example 1

```
JOB: PRDKPLO1 LIB CTM.PROD.SCHEDULE
                                                               TABLE: PRODKPL
COMMAND ===>
                                                               SCROLL===> CRSR
  OUT
  AUTO-ARCHIVE Y
                         SYSDB Y
                                         MAXDAYS
  RETENTION: # OF DAYS TO KEEP 030 # OF GENERATIONS TO KEEP
  SYSOUT OP (C,D,F,N,R)

MAXRERUN RERUNMEM

STEP RANGE FR (PGM.PROC)
                                                                      FROM
                                                    INTERVAL
                                                                   FROM
                                                        T0
  ON PGMST ANYSTEP PROCST UPDA CODES CO008 C0016
                                                                       A/0
    DO OK
    D0
  ON PGMST
                   PROCST
                                   CODES
                                                                       A/0
    D0
                     TIME
  SHOUT WHEN
                                 + DAYS
                                                                    URGN
 ----- >>>>>>>> END OF SCHEDULING PARAMETERS <<<<<<<<<<
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
                                                                      15.16.03
```

I

# **Example 2**

When procedure step UPDA in program step STEP08 finishes executing with a condition code less than C0008, it is considered OK.

Figure 250 ON PGMST Parameter – Example 2

JOB: PRDKPLO2 LIB CTM.PROD.SCHEDULE COMMAND ===>	TABLE: PRODKPL SCROLL===> CRSR
OUT	
AUTO-ARCHIVE Y SYSDB Y MAXDAYS MAXRUNS	
RETENTION: # OF DAYS TO KEEP 030 # OF GENERATIONS TO KEEP	FROM
SYSOUT OP (C,D,F,N,R) MAXRERUN RERUNMEM INTERVAL	FROM
STEP RANGE FR (PGM.PROC) . TO	•
ON PGMST STEP08 PROCST UPDA CODES <c0008< th=""><th>A/0</th></c0008<>	A/0
DO OK	
ON PGMST PROCST CODES	A/0
DO SHOULT HINE THE TO BLUE TO	UDON
SHOUT WHEN TIME + DAYS TO MS	URGN
====== >>>>>>>>>>>> END OF SCHEDULING PARAMETERS <<<<<	<<<<<<< ====
COMMANDS: EDIT, DOC, PLAN, JOBSTAT	15.16.03

## **Example 3**

When any program step in the step range DF2 (STEP20 – STEP29A) finishes executing with any system or user abend code, except U2030, rerun the job, and shout the indicated message to TSO logon ID P43.

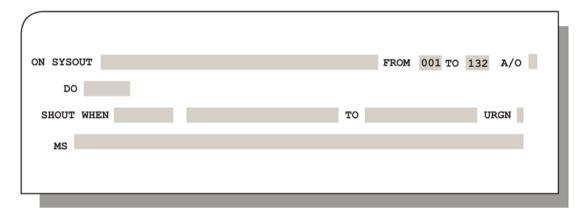
Figure 251 ON PGMST Parameter – Example 3

JOB: PRDKPLO3 L COMMAND ===>				TABLE: PRODKPL SCROLL===> CRSR
OUT				
	Y SYSDE	S Y MAXE	DAYS MAXRU	NS
RETENTION: #	OF DAYS TO KEEF	030 # OF GENE	ERATIONS TO KEE	P
SYSOUT OP (	C,D,F,N,R)			FROM
MAXRERUN	RERUNMEM		INTERVA	L FROM
	FR (PGM.PF		T0	
	PROCST	CODES S***	U**** NU20	30 A/0
DO RERUN				
	T0 TS0-P43		CY R	
	PLO3 ABENDED; TH	IF JOB 12 KEKUN		
DO ON DOMST	PROCST	CODES		A/0
DO	PRUCSI	CODE2		A/U
	TIME	+ DAYS	TO	URGN
MS	1 1112	57.10		o.r.g.r
====== >>>>>>	>>>>> ENG	OF SCHEDULING	PARAMETERS <<<	<<<<<<<<<=====
001111111111111111111111111111111111111	DOO DI IN			15 16 00
COMMANDS: EDIT	, DOC, PLAN, JOE	SSIAI		15.16.03

# **ON SYSOUT: Post-Processing Parameter**

Search for a string in the sysout of the job, and perform the actions in the accompanying DO statements if the string is found.

Figure 252 ON SYSOUT Parameter



Optional. Multiple ON SYSOUT parameters can be defined. Upon specifying an ON SYSOUT value and pressing **Enter**, a new ON SYSOUT statement, followed by a blank DO statement, is opened. Valid values are shown in Table 196.

Table 196 ON SYSOUT Values (part 1 of 2)

Value	Description
string	Mandatory if ON SYSOUT is used. The string to search for. This string may not consist of more than 40 alphanumeric characters. If you want this string to contain blanks, enclose the blanks, or the phrase containing the blanks, within single or double quotation marks.
	In addition:
	■ * – Use this character as a mask to match any string.
	&*- Use this to indicate an asterisk within a string where the asterisk is not a mask character.
	■ ? – Use this character as a mask to match any single character.
	&?- Use this to indicate a question mark within a string where the question mark is not a mask character.
FROM	The column in the SYSDATA files where the search must start. Valid values are from 1 through 132.

Table 196 ON SYSOUT Values (part 2 of 2)

Value	Description		
ТО	The column in the SYSDATA files where the search must end. Valid values are from 1 through 132.		
	ecommends the use of the FROM and TO parameters, if possible, to ed and thereby make the search more efficient.		
A/O	Optional. Boolean And/Or indicator. Valid values are:		
	■ A (And) – indicates AND logic between the two ON SYSOUT statements		
	ON SYSOUT block criteria are satisfied only if both ON SYSOUT statements are satisfied.		
	■ O (Or) – indicates OR logic between the two ON SYSOUT statements		
	ON SYSOUT block criteria are satisfied if either (or both) ON SYSOUT statements are satisfied.		
	If you specify A or O, an empty ON SYSOUT line is displayed.		
	Two or more ON SYSOUT statements connected by a Boolean indicator constitute an ON SYSOUT block.		
	The first DO statement is displayed after the last line of the ON SYSOUT block.		

# **General Information**

The ON SYSOUT parameter enables you to specify DO statements that are to be performed if the SYSDATA of the job contains a specified string. For this purpose, the SYSDATA of the job means the data contained in the following sysout files:

- JESMSGLG
- JESJCL
- JESYSMSG

If you know the location of the string, you can set values for the FROM and TO subparameters in order to restrict the search to a limited number of columns. This results in a more efficient search.

For a more advanced search, you can use the mask characters * and ? as described in Table 196.

Each ON SYSOUT line is checked against each line of the SYSDATA until a match is found or the end of the SYSDATA is reached. Each ON SYSOUT line is assigned a value of TRUE or FALSE.

Once the specified string is found, the following occurs:

- If there is only one ON SYSOUT line
  - Searching for that string ends.
  - The DO statement is performed.
- If an ON SYSOUT block has been specified, CONTROL-M
  - checks and assigns a value of TRUE or FALSE to each line
  - checks the value of the entire block to determine if it is TRUE or FALSE

If the result of that check is TRUE, the associated DO statements are performed.

## **Example 1**

ON SYSOUT IEF206I*STEP3	FROM 001 TO 050 A/O	
DO FORCEJOB		

CONTROL-M searches from Column 1 through Column 50 in each line for any string beginning IEF206I and ending STEP3.

## **Example 2**

ON SYSOUT IEF206I*&*STEP3	FROM 001 TO 050	A/0
DO FORCEJOB		

CONTROL-M searches from Column 1 through Column 50 in each line for any string beginning IEF206I and ending *STEP3.

### Example 3

ON SYSOUT 'IEF206I	STEP3'	FROM 001	T0	050	A/0
DO FORCEJOB					

The string IEF206I STEP3 contains a blank space, but is enclosed within quotation marks. CONTROL-M searches for the string from Column 1 through Column 50 in each line.

# **OUT: Post-Processing Parameter**

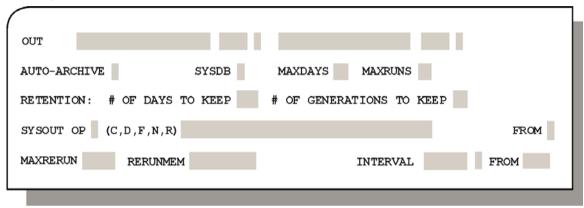
Add or delete prerequisite conditions when the job ends OK.

### **NOTE**



OUT and DO COND statements are similar. If you are familiar with one of them, you can easily use the other. However, familiarize yourself with the differences outlined below in "Differences between OUT and DO COND" on page 566.

Figure 253 OUT Parameter Format



Optional. A maximum of two prerequisite conditions can be specified in an OUT line. One prerequisite condition can be specified in each long OUT line. When you specify the second prerequisite condition in a standard OUT line, or one prerequisite condition in a long OUT line, and press **Enter**, a new OUT line is opened for specifying additional prerequisite conditions. For more information, see "Specifying Long OUT Condition Names" on page 565.

Each specified prerequisite condition consists of the mandatory subparameters described in Table 197.

Table 197 OUT Mandatory Subparameters (part 1 of 2)

Subparameter	Description
cond_name	User-supplied descriptive name of 1 through 39 characters used to identify the condition.
	<b>Note</b> : A condition name must not begin with the symbols " ", "¬", or "\", and must not contain parentheses (), because each of these characters has a special meaning.
	You can use an AutoEdit variable in a condition name, provided that the AutoEdit variable has a value that is known before the job is ordered.

Table 197 OUT Mandatory Subparameters (part 2 of 2)

Subparameter	Description
dateref	4-character date reference. Valid values are:
	<ul> <li>date – Specific date (in either mmdd or ddmm format, depending on the site standard).</li> </ul>
	■ ODAT – Resolves to the original scheduling date. Default.
	■ +nnn – Resolves at job order time to ODATE+nnn calendar days. nnn is three digits (000-999).
	■ -nnn – Resolves at job order time to ODATE-nnn calendar days. nnn is three digits (000-999).
	<b>Note:</b> +001 and -001 are not necessarily the same as NEXT and PREV, because NEXT and PREV are based on job scheduling criteria, while + <i>nnn</i> and - <i>nnn</i> are based on calendar days.
	<ul> <li>PREV – Resolves to the previous date on which the job ought to have been scheduled, according to its basic scheduling criteria (or ODATE-1 for a forced job).</li> </ul>
	■ NEXT – Resolves to the next date on which the job is scheduled according to its basic scheduling criteria (or ODATE+1 for a forced job).
	■ STAT – Static. Indicates that the condition, such as IMS-ACTIVE, is not date-dependent.
	Note: Before STAT was introduced, date 0101 was recommended to be used in conditions that were not date-dependent. Unlike 0101, STAT is not a date, and it operates differently. Always use STAT when defining conditions that are not date-dependent.
	**** – Any scheduling date. Valid only when <i>opt</i> is set to – .
	■ \$\$\$\$ – Any scheduling date. Valid only with <i>opt</i> is set to – .
	If a date reference is not specified, value ODAT is automatically inserted when you press <b>Enter</b> .
opt	Indicates whether to add or delete the specified prerequisite condition. Valid values are:
	<ul> <li>+ (Plus) - Add (create) the prerequisite condition</li> <li>- (Minus) - Delete the prerequisite condition</li> </ul>

## **General Information**

If the job ends OK, the prerequisite conditions are added to or deleted from the IOA Conditions file according to the value set for *opt*.

Prerequisite conditions are usually used to establish job dependencies or to ensure manual intervention when required:

■ To establish job dependency, define a prerequisite condition in an OUT or DO COND statement in the job that must run first, and in an IN statement in the job that must run afterwards

The job containing a prerequisite condition in an IN statement is not submitted unless that prerequisite condition has been added manually or by a job containing the OUT or DO COND statement.

- An OUT statement is used to add the prerequisite condition if the job ends OK.
- The DO COND statement is used to add the prerequisite condition if the step and code event criteria specified in the accompanying ON statement are satisfied.
- If the IN condition can only be satisfied by manual intervention, for example, if TAPE1-ARRIVED is set by the operator after an external tape has arrived on-site, performance of the required manual intervention before job submission can be ensured.

OUT and DO COND statements can also be used to delete prerequisite conditions. The OUT statement of the job can be used to delete a prerequisite condition after the job ends OK. A DO COND statement can be used to delete prerequisite conditions if the accompanying ON step and code criteria are satisfied.

These statements are generally used to delete prerequisite conditions either to prevent a particular job from running or when the condition is no longer needed by any other jobs in the Active Jobs file.

DO COND functions are performed after the functions of the OUT parameter:

- If a prerequisite condition is added by the OUT parameter and deleted by the DO COND parameter, the combined effect is the deletion of the prerequisite condition.
- If a prerequisite condition is deleted by the OUT parameter and added by the DO COND parameter, the combined effect is the addition of that prerequisite condition.

The following are examples of prerequisite conditions:

- IMS-ACTIVE
- JOB_PAYCALC_ENDED_OK
- TAPE1_LOADED

All prerequisite conditions are associated with a date reference that is used to distinguish between different runs of the same job with different scheduling dates. If, for example, a condition is being deleted, only the condition matching the specified date is deleted. The same condition from a different date is not deleted.

Prerequisite conditions created by the OUT parameter can trigger the execution of other jobs or processes.

Prerequisite conditions deleted by the OUT parameter can prevent the scheduling of jobs and processes that require those prerequisite conditions in their IN parameter.

For more information regarding prerequisite conditions, see "IN: Runtime Scheduling Parameter" on page 500, "ON Statements: Post–Processing Parameter" on page 537, and "DO COND: Post–Processing Parameter" on page 436, and see "Prerequisite Conditions" on page 69

# **Specifying Long OUT Condition Names**

Regular prerequisite conditions are not more than 20 characters in length. If you want to specify a longer condition name, up to 39 characters in length, enter the string **LONG** in the date reference field of an empty OUT condition line. An (L) appears at the beginning of the line. If the field already contains data, entering the string **LONG** will open a new long OUT condition line, with (L) appearing at the beginning of the line. You can now insert a long condition name, as illustrated in Figure 254.

Specify **SHRT** in the date reference field to revert back to condition names of standard length.





Long condition names cannot be used in CMEM rule definitions.

### Figure 254 Long OUT Condition

```
JOB: IEFBR14 LIB CTMP.V610.SCHEDULE
                                                         TABLE: PHILL1
COMMAND ===>
                                                         SCROLL ===> CRSR
 IN CTMLDNRS-NMIS-OK ODAT CTMLDNRS-NMIS-OK1 ODAT
        CTMLDNRS-NMIS-OK2 ODAT
 CONTROL CECI-ZEBRA-CONT E
                             0002
 RESOURCE INITOS
 PIPF
 FROM TIME 0800 + DAYS UNTIL TIME +
                                                  DAYS
 DUE OUT TIME + DAYS PRIORITY *1 SAC
                                                CONFIRM N
 TIME ZONE:
        CTMLDNRS-NMIS-OK ODAT - CTMLDNRS-NMIS-OK1 ODAT -
        (L) THIS-LINE-CONTAINS-A-LONG-OUT-CONDITION-XXXX ODAT -
 AUTO-ARCHIVE Y SYSDB Y MAXDAYS MAXRU RETENTION: # OF DAYS TO KEEP # OF GENERATIONS TO KEEP
                                         MAXDAYS MAXRUNS
 SYSOUT OP (C,D,F,N,R)
                                                                FROM
 MAXRERUN RERUNMEM
STEP RANGE FR (PGM.PROC) .
                                             INTERVAL
                                                             FROM
 ON PGMST ANYSTEP PROCST CODES ****
                                                                 A/0
   DO COND
   D0
 ON PGMST ANYSTEP PROCST
                               CODES ****
                                                                 A/0
COMMANDS: EDIT, DOC, PLAN, JOBSTAT
                                                                15.49.13
```

# **OUT Conditions for Group Entities**

Prerequisite conditions that are specified for Group entities in OUT statements and/or in ON GROUP-END DO COND statements enable you to establish dependencies between Group scheduling tables, and between Group scheduling tables and other jobs.

When all jobs in a Group scheduling table are ended or deleted, prerequisite conditions are added to or deleted from the IOA Conditions file according to the OUT statements and/or ON GROUP-END DO COND statements in the Group entity.

#### - NOTE



A Group entity can be reassigned a status of ACTIVE after specified prerequisite conditions have already been added or deleted. For example, if a job in the Group scheduling table was deleted while in WAIT SCHEDULE status and was then undeleted after the prerequisite conditions were added or deleted, the Group entity returns to ACTIVE status.

## **Differences between OUT and DO COND**

OUT and DO COND statements are similar but have the following differences:

■ An OUT statement is applied only if the job ends OK. DO COND statements are associated with ON statements and are applied only if the associated ON step and code criteria are satisfied.

- An OUT statement appears in each job scheduling definition. No DO COND statement appears unless specified. To specify a DO COND statement, type COND in an empty DO field and press Enter.
- DO COND statements are processed after OUT statements and can therefore override OUT statements.

# **Examples**

## **Example 1**

This example consists of two jobs (screens):

SACALCO1 - Calculates salaries

SARPT001 - Generates the Salary Statistics report

The report must be generated after the salaries have been successfully calculated.

Job SACALC01 runs first.

Figure 255 OUT Parameter Example 1 – First Job

```
JOB: SACALCO1 LIB CTM.PROD.SCHEDULE
                                                          TABLE: SALARY
COMMAND ===>
                                                          SCROLL===> CRSR
  CTB STEP AT NAME TYPE
  DOCMEM SACALCO1 DOCLIB CTM.PROD.DOC
  DAYS 01,15
                                                           DCAL
                                                                AND/OR
  WDAYS
                                                           WCAL
  MONTHS 1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- Y 11- Y 12- Y
  DATES
  CONFCAL SHIFT RETRO Y MAXWAIT OO D-CAT MINIMUM PDS
  DEFINITION ACTIVE FROM UNTIL
  CONTROL
  RESOURCE
  PIPE
 FROM TIME + DAYS UNTIL TIME + DAYS
DUE OUT TIME + DAYS PRIORITY SAC CONFIRM
TIME 70NF.
  TIME ZONE:
  OUT SALARY-OK ODAT +
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
```

When job SACALC01 ends OK, the prerequisite condition SALARY-OK is added. This triggered the execution of job SARPT001 that requires the condition in order to run.

Job SARPT001 is not run unless job SACALC01 ended OK.

Figure 256 OUT Parameter Example 1 – Second Job

```
JOB: SARPTOO1 LIB CTM.PROD.SCHEDULE
                                                              TABLE: SALARY
COMMAND ===>
                                                              SCROLL ===> CRSR
  SET VAR
  CTB STEP AT NAME
  DOCMEM SARPTOO1 DOCLIB CTM.PROD.DOC
  DAYS 01,15
                                                              DCAL
                                                                   AND/OR
  WDAYS
                                                              WCAL
  MONTHS 1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- Y 11- Y 12- Y
  DATES
  CONFCAL
                  SHIFT
                            RETRO Y MAXWAIT OO D-CAT
                  PDS
  DEFINITION ACTIVE FROM
                                UNTIL
          SALARY-OK
                             ODAT
  CONTROL
  RESOURCE
  PIPF
  FROM TIME + DAYS UNTIL TIME + DAYS
DUE OUT TIME + DAYS PRIORITY SAC CONFIRM
  TIME ZONE:
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
                                                                     11.17.00
```

The report (job SARPT001) is produced twice a month for the 1st and for the 15th. The report of the 15th is produced only if the prerequisite condition of the 15th, SALARY-OK, exists. The existence of the prerequisite condition of the 1st, SALARY-OK, does not cause submission of the report of the 15th (job SARPT001).

The jobs on the 1st, SACALC01 and SARPT001 (report), do not have to run on the 1st of the month. Suppose the salary job (SACALC01) finishes executing (OK) on the 3rd, the prerequisite condition SALARY-OK for the 1st is added, because the prerequisite condition is schedule date dependent.

### **Example 2**

Some jobs (such as IMSBDUPD) must run only when the IMS is active (IMS-ACTIVE):

MEMNAME	IMSDBUPD
DAYS	1,15
IN	IMS-ACTIVE STAT

The prerequisite condition IMS-ACTIVE is "generic," and it only exists when IMS is active. IMS itself is monitored by CONTROL-M. When IMS is brought down successfully, CONTROL-M deletes the prerequisite condition IMS-ACTIVE for all schedule dates. This prevents the abending of jobs that depend on IMS, such as job IMSDBUPD in the above example. Job IMSDBUPD is not submitted if the prerequisite condition IMS-ACTIVE does not exist.

Figure 257 OUT Parameter – Example 2

```
JOB: IMSPROD LIB CTM.PROD.SCHEDULE
                                                          TABLE: IMSPROD
COMMAND ===>
                                                          SCROLL==> CRSR
+-----
 CTB STEP AT NAME TYPE DOCMEM IMSPROD DOCLIB CTM.PROD.DOC
  DAYS
                                                           DCAL
                                                               AND/OR
  WDAYS
                                                           WCAL
  MONTHS 1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- Y 11- Y 12- Y
  CONFCAL
                 SHIFT
                          RETRO Y MAXWAIT 99 D-CAT
 MINIMUM
                 PDS
  DEFINITION ACTIVE FROM
                              UNTIL
  IN DEPOSITS
                             PRFV
  CONTROL
  RESOURCE
  PIPE
 FROM TIME + DAYS
DUE OUT TIME + DAYS
                               UNTIL TIME
                               PRIORITY SAC CONFIRM
  TIME ZONE:
          IMS-ACTIVE
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
```

### Example 3

A group of jobs runs every day of the week except Saturday and Sunday. Several of the different jobs for the different days must not run in parallel, and job sequence must be maintained even in case of delay.

### Figure 258 OUT Parameter – Example 3

```
JOB: EBDUPDT2 LIB CTM.PROD.SCHEDULE
                                                          TABLE: EBDPROD
COMMAND ===>
                                                          SCROLL ===> CRSR
  OVERLIB
                                                          STAT CAL
  SET VAR
  CTB STEP AT NAME
                                  TYPE
  DOCMEM EBDUPDT2 DOCLIB CTM.PROD.DOC
  DAYS
                                                           DCAL
                                                               AND/OR
  WDAYS 2,3,4,5,6
                                                           WCAL
  MONTHS 1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- Y 11- Y 12- Y
  DATES
                 SHIFT RETRO Y MAXWAIT 08 D-CAT
  CONFCAL
                 PDS
  MINIMUM
        DEPOSITS
                            PRFV
  ΙN
  CONTROL
  RESOURCE
  PIPE
  FROM TIME + DAYS UNTIL TIME + DAYS
DUE OUT TIME + DAYS PRIORITY SAC CONFIRM
                       ODAT +
  OUT DEPOSITS
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
                                                                11.17.00
```

The job is submitted only if the prerequisite condition DEPOSITS of the previous schedule date exists. If the job finishes OK, the prerequisite condition DEPOSITS of the same schedule date is added. This, in turn, triggers the next job in the sequence.

## **Example 4**

The following example serves as a further explanation of the schedule date reference concept:

```
MEMNAME EBDUPDT2

MEMLIB EBD.PROD.JOB

DAYS 1,15,20

MONTHS 1- N 2- N 3- N 4- N 5- N 6- N 7- Y 8- N 9- Y 10- N 11- Y 12- N

OUT EBD-INPUT-READY **** -
```

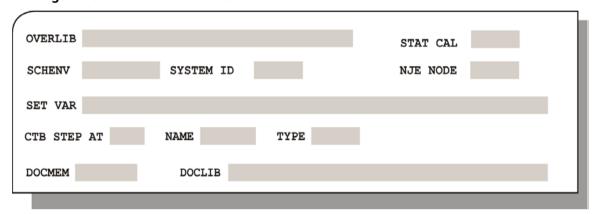
Today is September 15. The date reference values resolved in this job are in mmdd date format:

- ODAT 0915
- PREV 0901
- NEXT 0920
- **** Any date reference

# **OVERLIB: General Job Parameter**

Name of a JCL library that can override the library specification in the MEMLIB parameter, which is discussed on page 523.

Figure 259 OVERLIB Parameter Format



Optional. Valid values are:

- a valid data set name of 1 through 44 characters. AutoEdit variables can be specified.
- the reserved value DUMMY (for dummy jobs).

## **General Information**

The OVERLIB parameter enables submission of a modified copy of the actual JCL of the job, without changes to either the regular JCL (in the MEMLIB library) or the job scheduling definition.

The library containing the regular JCL member of the job is specified in the MEMLIB parameter. When temporary changes are desired, the JCL member can be copied to the library specified in the OVERLIB field and modified as needed.

# __ NOTE



When copying the regular JCL member to the OVERLIB library, do not change the member name. CONTROL-M always looks for a JCL member whose name matches the MEMNAME value.

If the MEMNAME member is found in the OVERLIB JCL library, that member is used. Otherwise, the member is taken from the MEMLIB library.

When the job is scheduled, the OVERLIB field is scanned. If it is not empty, CONTROL-M resolves AutoEdit variables in the field, if any are specified, and then searches the OVERLIB library for the member specified in field MEMNAME.

The override can be canceled by deleting the MEMNAME member from the OVERLIB library. If the MEMNAME member is not found in the OVERLIB library, the member is taken from the MEMLIB library. Alternatively, the override can be canceled by deleting the OVERLIB specification from the job scheduling definition.

### - NOTE -



OVERLIB cannot be specified for a started task.

The library can be any cataloged, standard partitioned data set, LIBRARIAN or PANVALET. The record length must be 80.

GENERAL or USER=*name*, which are valid MEMLIB values, cannot be specified in field OVERLIB.

The library and the member do not have to exist when the OVERLIB parameter is defined. Their existence is checked by CONTROL-M before actual submission of the job.

If, during the access to a library by CONTROL-M (before submission) the library is held exclusively by another user (such as TSO user, job), the monitor re-tries to access the library every few seconds until the library is released and the job can be submitted.

Use of the library name DUMMY is intended for scheduling events (for example, adding a prerequisite condition without actually running the job). If the library name DUMMY is used, the job is not submitted (submission and sysout checking is skipped). The job is assumed to have ended OK; ON PGMST...DO processing is not performed. All Post-processing parameters associated with an ENDED OK status are activated (OUT, SHOUT WHEN OK, and so on).

Three optional functions that were performed by the CTMX015C and CTMX015O exits in previous versions are now incorporated into the CONTROL-M monitor. These functions are controlled by the following installation parameters:

- COPMEM2O Copy the JCL member from the MEMLIB library to the OVERLIB library if the job ended NOTOK.
- DELOVRER Delete the JCL member from the OVERLIB library if the rerun of the job ended OK.
- DELOVRUN Delete the JCL member from the OVERLIB library if any run of the job ended OK

For a description of these parameters, see the chapter that discusses customizing INCONTROL products in the *INCONTROL for z/OS Installation Guide*.

# **Editing A Member through The J (JCL) Option**

### - NOTE -



You can only perform this function in an ISPF environment.

When specifying option J (JCL) in the Job List screen or the Active Environment screen to edit the JCL member, CONTROL-M must determine which library (MEMLIB or OVERLIB) to use.

The algorithm for this decision depends on:

- where the member exists, that is, whether it is only in the MEMLIB library, only in the OVERLIB library, in both libraries, or in neither library)
- what CTMIMACx REXX EXECs (if any) are defined
- from which screen the J (JCL) option was requested

Table 198 indicates which libraries are used, depending on the above criteria.

Table 198 OVERLIB Parameter: Algorithm for Libraries Used when Option J (JCL) is Specified (part 1 of 2)

When the following CTMIMAC is defined (if any)	the member exists in either library MEMLIB, OVERLIB, both, or neither	and the screen of the J (JCL) request is: Job List, Active Environment, or either screen	and the edit is performed in the following library
None (default)	MEMLIB only	Either screen	MEMLIB
	OVERLIB only	Either screen	OVERLIB
	Both	Either screen	OVERLIB
	Neither	Either screen	MEMLIB

Table 198 OVERLIB Parameter: Algorithm for Libraries Used when Option J (JCL) is Specified (part 2 of 2)

When the following CTMIMAC is defined (if any)	the member exists in either library MEMLIB, OVERLIB, both, or neither	and the screen of the J (JCL) request is: Job List, Active Environment, or either screen	and the edit is performed in the following library
CTMIMAC1	MEMLIB only	Job List	MEMLIB
		Active Environment	OVERLIB (copied from MEMLIB)
	OVERLIB only	Job List	MEMLIB (open empty member)
		Active Environment	OVERLIB
	Both	Job List	MEMLIB
		Active Environment	OVERLIB (not copied)
	Neither	Job List	MEMLIB
		Active Environment	OVERLIB
CTMIMAC2	MEMLIB only	Either screen	MEMLIB
	OVERLIB only	Either screen	OVERLIB
	Both	Either screen	OVERLIB
	Neither	Either screen	OVERLIB
CTMIMAC3	MEMLIB only	Job List	MEMLIB (But saved only if changed)
		Active Environment	OVERLIB
	OVERLIB only	Job List	MEMLIB (open empty member)
		Active Environment	OVERLIB
	Both	Job List	MEMLIB
		Active Environment	OVERLIB
	Neither	Job List	MEMLIB
		Active Environment	OVERLIB

## Note the following points:

■ When using CTMIMAC1 or CTMIMAC2, more than four libraries cannot be concatenated.

- When using CTMIMAC3, Exit CTMX014G, in the IOA SAMPEXIT library, is required if libraries are concatenated.
  - For concatenated libraries, if the &COPYMEM parameter in Exit CTMX014G is set to YES, the saved member is placed in the first library of concatenation. If the &COPYMEM parameter is set to NO, the saved member is placed back in the original JCL library.
- The CTMIMACx REXX EXECs can be found in the IOA CLIST library. Instructions for installing these REXX EXECs can be found in comments in the members themselves.
- PANVALET and LIBRARIAN considerations when performing online JCL edits:
  - For PANVALET or LIBRARIAN support, sample exits CTMX014P or CTMX014L, respectively, must be installed. However, CONTROL-M does not support both products simultaneously.
  - When both MEMLIB and OVERLIB exist, and MEMLIB is either PANVALET or LIBRARIAN, the edit function first copies the member to the OVERLIB before performing the edit, unless the member already exists in the OVERLIB.
  - IF only MEMLIB exists, and it is LIBARIAN, the edit is performed directly in MEMLIB.
  - If the MEMLIB library is PANVALET, editing can only be performed if a non-PANVALET OVERLIB is defined.

# **Example**

If a special, modified version of BACKPL02 JCL is required, it is defined in CTM.OVER.JOBLIB. This JCL is used instead of the JCL in CTM.PROD.JOBLIB.

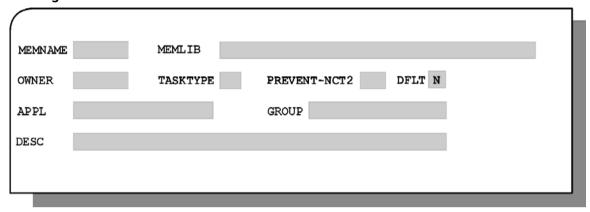
### Figure 260 OVERLIB Parameter Example

```
JOB: BACKPLO2 LIB CTM.PROD.SCHEDULE
                                                       TABLE: BACKUP
COMMAND ===>
                                                       SCROLL===> CRSR
+------
  MEMNAME BACKPLO2 MEMLIB CTM.PROD.JOBLIB
  OWNER M44 TASKTYPE JOB PREVENT-NCT2 Y DFLT N
  APPL APPL-L
                                 GROUP BKP-PROD-L
  DESC
        DAILY BACKUP OF SPECIAL FILES FROM APPL-L
  OVERLIB CTM.OVER.JOBLIB
                                                     STAT CAL
  SCHENV
                            SYSTEM ID
                                                     NJE NODE
  SET VAR
  CTB STEP AT NAME TYPE
DOCMEM BACKPLO2 DOCLIB CTM.PROD.DOC
  DAYS
                                                        DCAL
                                                            AND/OR
                                                        WCAL
  MONTHS 1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- Y 11- Y 12- Y
  CONFCAL WORKDAYS SHIFT
                          RETRO N MAXWAIT 04 D-CAT
  MINIMUM
               PDS
  DEFINITION ACTIVE FROM
                             UNTIL
         START-DAILY-BACKUP ODAT
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
                                                             11.17.00
```

# **OWNER: General Job Parameter**

User who requests CONTROL-M services.

Figure 261 OWNER Parameter Format



Mandatory. OWNER must be 1 through 8 characters.

### **General Information**

The OWNER parameter is used by the internal security mechanism of CONTROL-M to determine which operations each user is authorized to perform and which information each user is authorized to access. For example, access to options and information on the Active Environment screen can be limited by the OWNER parameter.

The OWNER parameter can also facilitate selection and handling of production jobs.

The OWNER parameter is passed to external security products, such as RACF, ACF2 and TOP SECRET. Certain security products require that the owner name not exceed seven characters.

Default OWNER is dependent on the online environment of the site (CICS, TSO, and so on). For TSO and TSO/ISPF environments, the TSO user ID is the default. For non-TSO environments, such as CICS, the default is the terminal ID.

# **Example**

Job OPERCOMP belongs to owner SYS1.

#### Figure 262 OWNER Parameter Example

```
JOB: OPERCOMP LIB CTM.PROD.SCHEDULE
                                                                 TABLE: OPER
COMMAND ===>
                                                                 SCROLL===> CRSR
+----
  MEMNAME OPERCOMP MEMLIB GENERAL

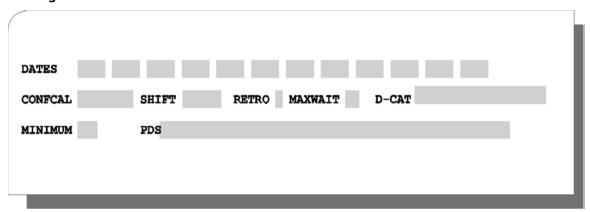
OWNER SYS1 TASKTYPE JOB PREVENT-NCT2 DFLT N

APPL OPER GROUP MAINTENANCE
  APPL OPER
                                      GROUP MAINTENANCE
  DESC JOB RUN ON THE 1ST OF THE MONTH
  OVERLIB
                                                             STAT CAL
  SCHENV
                                SYSTEM ID
                                                             NJE NODE
  SET VAR
  CTB STEP AT NAME TYPE
DOCMEM OPERCOMP DOCLIB CTM.PROD.DOC
                                                                DCAL
  DAYS 01
                                                                      AND/OR
  WDAYS
                                                                  WCAL
  MONTHS 1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- Y 11- Y 12- Y
  DATES
  CONFCAL
                   SHIFT
                              RETRO Y MAXWAIT OO D-CAT
  MINIMUM
                   PDS
  DEFINITION ACTIVE FROM
                                  UNTIL
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
                                                                       11.17.00
```

# PDS: Basic Scheduling Parameter

Partitioned data set (library) that must be checked for the minimum number of required free tracks. That number is specified in the MINIMUM parameter, which is described on page 531.

Figure 263 PDS Parameter Format



Optional; however, if MINIMUM is specified, PDS is mandatory. The PDS parameter specifies a data set name of 1 through 44 characters.

The PDS parameter cannot be used with any of the following parameters: DAYS, WDAYS, MONTHS, CONFCAL, RETRO and DATES.

### **General Information**

The data set must be cataloged, and it must be a partitioned data set.

The MINIMUM and PDS parameters are always used together and are never used with other Basic Scheduling parameters.

The PDS parameter identifies a library. The MINIMUM parameter specifies the minimum number of free tracks required by that library.

These parameters are intended for use (that is, definition) in jobs or started tasks that compress, clean and/or enlarge libraries, or which issue a warning message to the IOA Log file, that is, if the TASKTYPE parameter is set to WRN.

If the MINIMUM and PDS parameters are defined for a job, the scheduling of the job is not related to or dependent upon any date criteria. Instead, the job is scheduled if the actual number of free tracks available in the specified library is below the specified minimum when the New Day procedure is run. The job or started task can then compress, clean, or enlarge the library, or issue the appropriate warning.





The PDS parameter does not work with PDSE-type libraries because they always appear to be 100 percent full.

# **Example**

Check the SYS1.LINKLIB library for a minimum of 20 unused tracks.

#### Figure 264 PDS Parameter Example

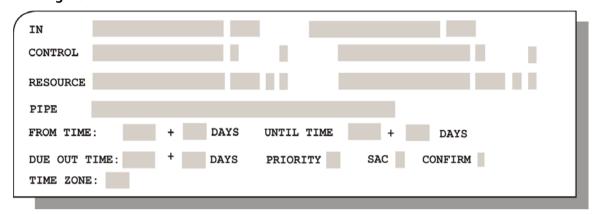
```
JOB: MSG001 LIB CTM.PROD.SCHEDULE
                                                                  TABLE: OPER
COMMAND ===>
                                                                  SCROLL===> CRSR
  MEMNAME MSG001 MEMLIB GENERAL
OWNER SYS1 TASKTYPE WRN PREVENT-NCT2 DFLT N
APPL OPFR GROUP OPER-MAINT
  APPL OPER GROUP OPER-MAINT DESC INDICATE COMPRESS IS NEEDED FOR SYS1.LINKLIB
                                                               STAT CAL
  SCHENV
                                  SYSTEM ID
                                                               NJE NODE
  SET VAR
  CTB STEP AT NAME TYPE
DOCMEM MSG001 DOCLIB CTM.PROD.DOC
  DAYS
                                                                   DCAL
                                                                    AND/OR
                                                                   WCAL
  WDAYS
  MONTHS 1- 2- 3- 4- 5- 6- 7- 8- 9- 10-
  DATES
                   SHIFT RETRO Y MAXWAIT OO D-CAT
  CONFCAL
  MINIMUM 020 PDS SYS1.LINKLIB
  DEFINITION ACTIVE FROM UNTIL
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
```

The MSG001 member in the CONTROL-M GENERAL JCL library contains a warning message to compress the library.

# **PIPE: General Job Parameter**

Indicates a data set to be replaced by a pipe with the same name. Displayed only if MAINVIEW Batch Optimizer (MVBO) is installed.

Figure 265 PIPE Parameter Format



Optional. Valid value is a valid data set name (1 through 44 characters).

Each time a data set or pipe name is specified and **Enter** is pressed, a new empty line is displayed to allow specification of an additional data set or pipe name.

### **General Information**

Pipes are storage buffers that are used to replace data sets. Pipes are defined in, and used by, MVBO/Job Optimizer Pipes to replace sequential processing with parallel processing.

For example, normally, that is, without pipes, if JOB1 writes to data set DS1 and then JOB2 reads data set DS1, JOB2 waits until JOB1 is terminated before reading the data set. However, if a pipe is used to replace data set DS1, then as JOB1 writes data to pipe DS1, JOB2 can use the data without waiting for termination of JOB1.

Each pipe and its relevant parameters are defined in a MBVO/Job Optimizer Pipes rule. Each pipe must be defined with the same name as the data set it is replacing.

When a job is to use a pipe instead of a data set, the name of the data set or pipe must be specified in the PIPE parameter of the CONTROL-M job scheduling definition for the job.

For more information about Pipe processing, see "Job-Related Considerations for Pipes" on page 821

# **Example**

This example consists of two job scheduling definitions.

In job CTLIVPWR and job CTLIVPRD, data set CTL.IVP.FILE is replaced by a pipe with the same name. Jobs such as CTLIVPWR below and CTLIVPRD on page 583 are called a "Collection" because they are pipe participants of the same pipe.

Figure 266 PIPE Parameter Example – Job CTLIVPWR

```
JOB: CTLIVPWR LIB CTMT.PROD.SCHEDULE
                                                             TABLE: CTLIVP
COMMAND ===>
                                                             SCROLL===> CRSR
+-----
  MEMNAME CTLIVPWR MEMLIB CTM.IVP.JCL
OWNER E02A TASKTYPE JOB PREVENT-NCT2 DFLT N
  APPL
                                    GROUP
        MAINVIEW BATCH OPTIMIZER VERIFICATION - WRITER JOB
  DESC
  OVERLIB
                                                         STAT CAL
  SCHENV
                               SYSTEM ID
                                                         NJE NODE
  SET VAR
  CTB STEP AT NAME
                                   TYPE
  DOCMEM CTLIVPWR DOCLIB CTMT.PROD.DOC
  DAYS
                                                                 AND/OR
  WDAYS
                                                             WCAL
  MONTHS 1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- Y 11- Y 12- Y
  DATES
  CONFCAL
                  SHIFT
                             RETRO N MAXWAIT 00 D-CAT.
  MINIMUM
                  PDS
  DEFINITION ACTIVE FROM
                                UNTIL
          CTLIVPWR-IN
                             ODAT
  ΤN
  CONTROL
  RESOURCE
  PIPE
          CTL.IVP.FILE
  PIPE
                        DAYS UNTIL TIME +
  FROM TIME
                                                    DAYS
  DUE OUT TIME
                        DAYS
                                PRIORITY SAC CONFIRM
  TIME ZONE:
```

Ì

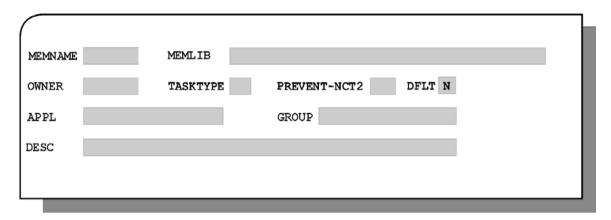
#### Figure 267 PIPE Parameter Example – Job CTLIVPRD

```
JOB: CTLIVPRD LIB CTMT.PROD.SCHEDULE
                                                           TABLE: CTLIVP
COMMAND ===>
                                                           SCROLL===> CRSR
  MEMNAME CTLIVPRD MEMLIB CTM.IVP.JCL
  OWNER E02A TASKTYPE JOB PREVENT-NCT2 DFLT N
                                   GROUP
  DESC MVBO VERIFICATION - READER JOB
  OVERLIB
                                                        STAT CAL
  SCHENV
                             SYSTEM ID
                                                        NJE NODE
  SET VAR
  CTB STEP AT NAME TYPE
DOCMEM CTLIVPRD DOCLIB CTMT.PROD.DOC
  DAYS
                                                            DCAL
                                                             AND/OR
                                                            WCAL
  WDAYS
  MONTHS 1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- Y 11- Y 12- Y
  DATES
  CONFCAL
                 SHIFT RETRO N MAXWAIT 00 D-CAT
  MINIMUM
  DEFINITION ACTIVE FROM
                              UNTIL
  IN CTLIVPWR-OUT ODAT
  CONTROL
  RESOURCE
  PIPE CTL.IVP.FILE
  PIPE
  FROM TIME + DAYS UNTIL TIME + DAYS
DUE OUT TIME + DAYS PRIORITY SAC CONFIRM
  TIME ZONE:
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
                                                                  13.22.07
```

# §Restart§PREVENT-NCT2:General Job Parameter

Performs data set cleanup before the original job run.

Figure 268 §Restart§ PREVENT-NCT2 Parameter Format



Optional. PREVENT-NCT2 consists of the subparameters described in Table 199.

Table 199 §Restart§ PREVENT-NCT2 Subparameters

Subparameter	Description
PREVENT-NCT2	Whether, and how, to perform data set cleanup before the original run of the job. Optional. Valid values are:
	<ul> <li>Y (Yes) - Perform data set cleanup before the original job run; this value is not valid for started tasks</li> </ul>
	■ N (No) – Do not perform data set cleanup before the original job run
	■ F (Flush) – Halt processing of the job if any data set cleanup error is detected, even if MVS would not have stopped processing the job
	■ L (List) – Do not perform data set cleanup before the original job run; but generate the messages that would be required for GDG adjustment during restart
DFLT	Protected field indicating the PREVENT-NCT2 default value for the site. The default is set in parameter NCAT2 in the CTRPARM member in the IOA PARM library. A value specified in the PREVENT-NCT2 parameter overrides the site default.

### **General Information**

If a job tries to create a data set that already exists, the job may fail with a DUPLICATE DATASET ON VOLUME error. If a job tries to create a data set whose name is already cataloged, the job may fail with an error message that indicates a reason of NOT CATLGD for reason code 2; the CONTROL-M/Restart term PREVENT-NCT2 is derived from this error situation.

These problems can be avoided by performing data set cleanup. During data set cleanup, CONTROL-M/Restart does the following:

- Deletes and uncatalogs the old data sets. This prevents DUPLICATE DATSET ON VOLUME and NOT CATLGD 2 errors.
- Performs Generation Dataset (GDG) Adjustment, which is described in the *CONTROL-M/Restart User Guide*

CONTROL-M/Restart automatically performs data set cleanup prior to restarts and reruns. However, it may be desirable to perform data set cleanup before the original job run, because data sets accessed by the job can have file-related errors that were generated by an entirely different job.

When data set cleanup is performed as part of the original job request, it is called PREVENT-NCT2 processing.

The site-defined default in NCAT2 in the CTRPARM member determines whether data set cleanup is to be performed before the original job run. The value of this site-defined default is displayed in protected field DFLT.

The PREVENT-NCT2 parameter can be used to override this default to determine what data set cleanup instructions are provided to the original job run. Possible values, and their effects, are described below:

■ When value Y is specified:

CONTROL-M/Restart performs data set cleanup before the original job run. It deletes and uncatalogs all data sets that can cause NCT2 and duplicate data set errors during execution, and performs GDG adjustment if necessary.

When value F is specified:

If a file catalog error is detected, processing is halted, even if normal MVS processing would not handle the problems as a fatal error, and an appropriate error message is generated.

■ When value L is specified:

Data set cleanup is not performed for the original run, but messages that would be required for GDG adjustment during restart are generated. Without these messages, GDG adjustment might not be properly performed during restart. In addition to the GDG adjustment messages, the same messages that are generated during simulation of data set cleanup are also generated.

#### - NOTE



If you would normally specify N, meaning CONTROL-M/Restart processing is not desired for the original run, but the JCL requires GDG processing, it is recommended that you specify value L instead of value N.

■ When value N is specified:

No special action is taken by CONTROL-M/Restart. Data set cleanup is not performed.

If a value of Y, F, or L is specified, that is, if some kind of special NCT2 processing is desired, a CONTROLR step is automatically added as a first step of the submitted job.

The PREVENT NCT2 parameter has no impact on restarts, because CONTROL-M/Restart automatically performs data set cleanup prior to restarts.

#### - NOTE -



When PREVENT-NCT2 is active for a job, it is possible to automatically collect certain statistical information on the job. For more details, see the AUTOXREF parameter in the *INCONTROL for z/OS Installation Guide*.

# **Example**

Prevent NOT CATLGD 2 errors for job PRDKPL01.

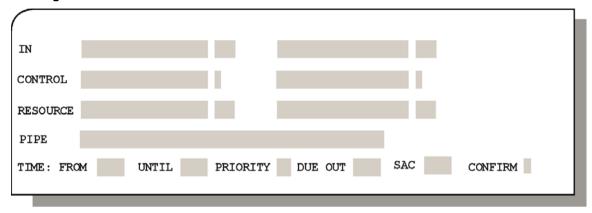
#### Figure 269 §Restart§ PREVENT-NCT2 Parameter Example

```
JOB: PRDKPL01 LIB CTM.PROD.SCHEDULE
                                                             TABLE: PRODKPL
                                                             SCROLL===> CRSR
+-----
 MEMNAME PRDKPL01 MEMLIB CTM.PROD.JCL
OWNER SYS1 TASKTYPE JOB PREVENT-NCT2 Y DFLT N
 APPL KPL
                                    GROUP PROD-KPL
 DESC DAILY PRODUCTION - START OF APPL-PROD-KPL
 OVERLIB
                                                          STAT CAL
  SCHENV
                               SYSTEM ID
                                                          NJE NODE
  SET VAR
  CTB STEP AT NAME
                                    TYPE
  DOCMEM PRDKPL01 DOCLIB CTM.PROD.DOC
                                                             DCAL
  DAYS 01
                                                                  AND/OR
  WDAYS
                                                              WCAL
  MONTHS 1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- Y 11- Y 12- Y
  DATES
  CONFCAL
                  SHIFT
                            RETRO Y MAXWAIT OO D-CAT
  MINIMUM
                  PDS
  DEFINITION ACTIVE FROM
                                UNTIL
          START-DAILY-PROD-KPL ODAT
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
                                                                  11.17.00
```

# **PRIORITY: Runtime Scheduling Parameter**

Internal CONTROL-M job/group priority.

Figure 270 PRIORITY Parameter Format



Optional. The PRIORITY parameter indicates a 1 to 2 character alphanumeric priority. An asterisk (discussed later) may also be specified.

The default is blank, which is the lowest priority.

### **General Information**

Priority helps determine the order in which jobs in the Active Jobs file are processed by CONTROL-M.

Priority is determined in ascending order where: blank < A < Z < 0 < 9 < *

In general, the job with the highest priority code executes first if all its other runtime scheduling requirements are satisfied.

When not all runtime requirements for a high priority job are satisfied, for example, where a job requires two tape drives but only one is available, a job with a lower priority whose other runtime requirements are satisfied may be run earlier.

This, however, is not always desirable. A job may be so important that lower priority jobs must not be submitted until the important job has executed.

Such a job is called a critical path job. Critical path priority can be indicated by prefixing the priority with an asterisk (*).

A priority prefixed by an asterisk, such as priority *5, indicates that CONTROL-M must submit the job before submitting any regular (non-*) priority jobs, such as priority 10, and before submitting any critical path jobs of lower priority, such as priority *3, even if the resources required for those other jobs are available.

Critical path priority is applied only after all the IN conditions for the job exist.

# Crit

#### NOTE -

Critical path priority applies to contention for Quantitative resources and for Control resources required in exclusive state. Critical path priority does not apply to contention for Control resources required in shared state.

# **Examples**

#### **Example 1**

The priority level of job EBDIN001 is 07, and it requires three tapes. The priority level of job EBDIN002 is 02, and it requires only one tape:

```
MEMNAME EBDIN001
RESOURCE TAPE 0003
PRIORITY 07
MEMNAME EBDIN002
RESOURCE TAPE 0001
PRIORITY 02
```

If only two tapes are available, job EBDIN002 is submitted.

#### **Example 2**

The priority level of job EBDUPDT is *5, and it requires two tapes. The priority level of job EBDEXEC is 04, and it requires one tape:

```
MEMNAME EBDUPDT
RESOURCE TAPE 0002
PRIORITY *5
MEMNAME EBDEXEC
RESOURCE TAPE 0001
PRIORITY 04
```

If one tape is available, neither job is submitted. When two tapes become available, job EBDUPDT is submitted.

### **Example 3**

The priority level of job EBDBKP is *8, and it requires three tapes. The priority level of job EBDMAINT is *7, and it requires one tape:

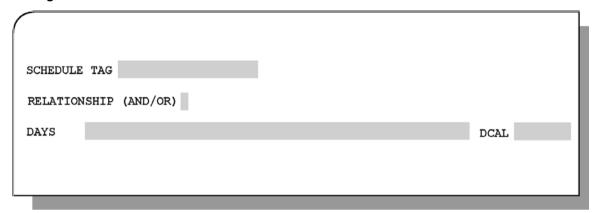
```
MEMNAME EBDBKP
RESOURCE TAPE 0003
PRIORITY *8
MEMNAME EBDMAINT
RESOURCE TAPE 0001
PRIORITY *7
```

If one tape is available, neither job is submitted. When three tapes become available, job EBDBKP is submitted.

# **RELATIONSHIP: Basic Scheduling Parameter**

Indicates the relationship (AND/OR) between schedule tag criteria and the basic scheduling criteria of the job, that is, whether either set of criteria, or both sets of criteria, are to be satisfied. This parameter appears in job scheduling definitions in Group tables only.

Figure 271 RELATIONSHIP Parameter Format



Optional. Valid values are described in Table 200.

**Table 200 RELATIONSHIP Parameter Values** 

Value	Description
O (Or)	If either set of criteria (a schedule tag or the basic scheduling criteria of the job) are satisfied, the job is scheduled. Default.
A (And)	Both a schedule tag and the basic scheduling criteria of the job must be satisfied.

## **General Information**

For jobs in Group scheduling tables, two types of basic scheduling criteria can be specified:

**Table 201 RELATIONSHIP Parameter Scheduling Types** 

Туре	Description
Schedule Tags	Pointers to sets of Group criteria (that is, basic scheduling criteria defined for the table in the Group Entity).
Basic scheduling	Basic scheduling criteria defined in, and belonging to, the job scheduling definition. They are not connected to Group criteria.

In some cases, it may be required that both sets of criteria be satisfied. In other cases, satisfaction of either set of criteria is sufficient for job scheduling. This parameter allows specification of the required combination:

- When either set of criteria is sufficient, specify value O (OR relationship).
- When both sets of criteria are required, specify value A (AND relationship).

If an AND relationship is specified when no schedule tags are defined in the job, the job is never scheduled.

For more information, see Figure 137 on page 384.

# **Example**

Create a table of employee hours each payday and on the last day of the year.

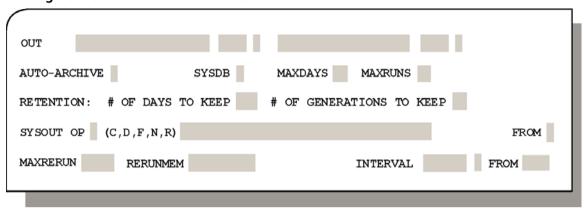
Figure 272 RELATIONSHIP Parameter Example

```
JOB: TABHOURS LIB CTM.PROD.SCHEDULE
                                                             TABLE: ACCOUNTS
COMMAND ===>
                                                             SCROLL===> CRSR
  MEMNAME TABHOURS MEMLIB CTM.PROD.JCL
  OWNER NO4B TASKTYPE JOB PREVENT-NCT2 DFLT N
  APPL
                                   GROUP ACCOUNT_GROUP
        TABULATE EMPLOYEE HOURS
  DESC
  OVERLIB
                                                            STAT CAL
  SET VAR
  CTB STEP AT
                   NAME
                                   TYPF
  DOCMEM TABHOURS DOCLIB CTM.PROD.DOC
  SCHEDULE TAG PAYDAYS
  SCHEDULE TAG
  RELATIONSHIP (AND/OR) 0
  DAYS
         31
                                                             DCAL
                                                                 AND/OR
  WDAYS
  MONTHS 1- N 2- N 3- N 4- N 5- N 6- N 7- N 8- N 9- N 10- N 11- N 12- Y
  DATES
                  SHIFT
                             RETRO N MAXWAIT OO D-CAT
  CONFCAL
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
                                                                   18.36.07
```

# **RERUNMEM: Post-Processing Parameter**

Name of the JCL member to use when the job is automatically rerun. (Called RERUN – RERUNMEM prior to version 6.0.00.)

Figure 273 RERUNMEM Parameter Format



Optional. The RERUNMEM parameter identifies a valid member name of 1 through 8 characters.

### **General Information**

Although the RERUNMEM parameter can be used to specify the name of a JCL member to use for automatic rerun, note the following points:

- The DO FORCEJOB parameter provides a more flexible alternative to the RERUNMEM parameter.
- CONTROL-M/Restart users can use the DO IFRERUN parameter to restart the failed job instead of using RERUNMEM to rerun the job.

The automatic rerun process works as follows:

■ The CONTROL-M monitor determines that automatic rerun is possible only if the job ENDS NOTOK and a specified DO RERUN statement is activated during post-processing. If the monitor determines that automatic rerun is possible, it sets the status of the job to ENDED NOTOK – RERUN NEEDED.

- The monitor then checks the value of MAXRERUN in the Active environment. If the value is zero, or no MAXRERUN value was specified, automatic rerun is not possible and the job is not submitted for rerun. If the value is greater than zero, rerun is possible, and the monitor submits the job for rerun when all runtime criteria are satisfied. Runtime criteria include not only the Runtime Scheduling parameters, but also the INTERVAL parameter, which specifies the minimum allowable interval between runs of the same job.
- The JCL for the rerun job is taken from the member specified in the RERUNMEM parameter. If no RERUNMEM value is specified, the JCL for the rerun is taken from the regular JCL member of the job specified in the MEMNAME parameter.

Rules applying to the MEMNAME parameter also apply to the RERUN parameter.

The member name can be the same as, or different from, the job name.

The member specified in RERUNMEM must be in the library specified in the MEMLIB parameter.

The RERUNMEM parameter overrides the MEMNAME value in the JCL, and the MEMNAME value becomes irrelevant for reruns.

The RERUNMEM parameter cannot be specified for cyclic jobs and cyclic started tasks.

The RERUNMEM parameter cannot be specified if a DO IFRERUN statement is specified.

# **Example**

If job EF145TS abends during step name COLLECT, try to run another job from the member EF145TSR that continues from the same place.

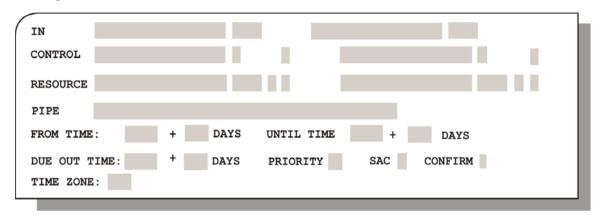
Figure 274 RERUNMEM Parameter Example

JOB: EF145TS LIB CTM.PROD.SCHEDULE COMMAND ===>	TABLE: EFPROD SCROLL===> CRSR
+	+
OUT	
AUTO-ARCHIVE Y SYSDB Y MAXDAYS MAXRUI	NS
RETENTION: # OF DAYS TO KEEP 030 # OF GENERATIONS TO KEEP	P
SYSOUT OP (C,D,F,N,R)	FROM
MAXRERUN 2 RERUNMEM EF145TSR INTERVAL	FROM
STEP RANGE FR (PGM.PROC) . TO ON PGMST COLLECT PROCST CODES S*** U****	
DO RERUN	A/0
DO REROW	
ON PGMST PROCST CODES	A/0
DO	
SHOUT WHEN TIME + DAYS TO	URGN
MS	
>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>	<<<<<<<< =====
COMMANDS: EDIT, DOC, PLAN, JOBSTAT	11.17.00

# **RESOURCE: Runtime Scheduling Parameter**

Quantitative resources and their quantities required by the job.

Figure 275 RESOURCE Parameter Format



Optional. A maximum of two Quantitative resources can be specified in each RESOURCE line. Upon specifying the second Quantitative resource in a line and pressing **Enter**, a new line is opened, for specifying additional Quantitative resources.

Each specified Quantitative resource consists of the mandatory subparameters described in Table 202 and may include the optional parameters described in Table 204.

**Table 202 Mandatory RESOURCE Subparameters** 

Subparameter	Description
res_name	User-supplied, descriptive name of 1 through 20 characters to identify the Quantitative resource.
quantity	Quantity of the Quantitative resource required by the job. The specified value must be four digits (leading zeros must be specified).

Table 203 Optional RESOURCE Subparameters (part 1 of 2)

Subparameter	Description
onOk	Whether to keep the Quantitative resource tied to the job if the job ends OK. Valid values are:
	<ul> <li>' '(blank) – release the resource</li> <li>The resource is not kept, and is returned to the total quantity available for other jobs. Default.</li> </ul>
	■ D – discard the resource The resource is not reusable. The quantity of the resource is permanently removed from the total quantity available for other jobs.

Table 203 Optional RESOURCE Subparameters (part 2 of 2)

Subparameter	Description
onFail	Whether the Quantitative resource remains allocated to the job if the job ends NOTOK. Valid values are:
	■ ' '(blank) – release the resource The resource does not remain allocated to the job. Default.
	<ul> <li>K - keep the resource</li> <li>The resource remains allocated to the job until one of the following occurs:</li> <li>the job is rerun and ends OK</li> <li>the job is deleted</li> <li>the job is FORCED OK</li> </ul>

#### **Example**

TAPE 2 D K

Assume that the total number of TAPE resources is 5.

If the job ends OK, the total number of TAPE resources stays at 3, since 2 TAPE resources are permanently discarded.

If the job is rerun, and there are enough TAPE resources, CONTROL-M allocates a quantity of 2 TAPE resources for the job. The total number of TAPE resources is then reduced to 1.

If the job ends NOTOK, the total number of TAPE resources stays at 5, but the available number of resources will be 3. The job keeps the 2 TAPE resources.

If the job is rerun, CONTROL-M reallocates a quantity of 2 TAPE resources for the job. The total number of TAPE resources stays at 5, and the available number of resources stays at 3.

### **General Information**

Quantitative resource specification can be used to prevent resource contention.

Quantitative resources, such as tape drives, CPU, and access rates to the spool, and their maximum available quantities are defined for the site in the IOA Conditions/Resources screen (Screen 4).

#### - NOTE

If the AUTOTAPE parameter in the CTMPARM member in the IOA PARM library is set to Y, any tape drive value defined in the RESOURCE parameter is ignored. Instead, a value determined by the Automatic Tape Adjustment facility is used. For more information, see "Tape Device Usage Statistics" on page 241, and the description of using the Automatic Tape Adjustment facility in the *INCONTROL for z/OS Administrator Guide*.

To remove selected Quantitative resources from job scheduling definitions, use the CTMTBUPD utility described in the *INCONTROL* for *z/OS Utilities Guide*.

# **Examples**

TAPE 12 CPU 80 WORK-SPACE 3000

To eliminate bottlenecks and maximize throughput, a site can quantify processing power and assign it a resource name, such as CPU or LPU (logical processing units). The more powerful the CPU, the greater the maximum quantity that can be assigned to it.

The RESOURCE parameter is used to specify the quantity of a resource required by the job.

Before a job is submitted, CONTROL-M verifies that the required quantities of resources, defined through RESOURCE statements, are available, that is, that they are not in use by another job:

- If they are available, CONTROL-M allocates them to the job, and they become unavailable to other jobs until they are freed.
- If the resources required by the job are unavailable, the job is not submitted.

#### **Resource Allocation in Multi-CPU Environments**

In multi-CPU environments, several resource allocation possibilities exist.

One possibility is to operate as if there is one large CPU and resource pool. In this case, no logical differentiation between CPUs is made, and the CONTROL-M monitor assigns resources, including CPU processing power, from the total resources available.

Another possibility is to differentiate between CPUs and optionally to logically associate quantities of resources with specific CPUs.

This is generally achieved through the use of common identifiers, such as a suffix.

For example, suppose a site has three CPUs of differing processing capability. The following representative resources and quantities might be defined in the IOA Resources file:

CPU-A 50 CPU-B 75 CPU-C 100

In this example, it might also be desired to logically categorize other resources according to CPU. For example, if 12 tape drives are available, the following resources and quantities might also be defined in the IOA Resources file:

TAPE-A 3
TAPE-B 4
TAPE-C 5

If this kind of differentiation is used, different resources in the job scheduling definition can be specified with different suffixes, and the job still runs. For example, a quantity of CPU-A can be specified along with a quantity of TAPE-B.

Rather than specifying a particular identifier when requesting a resource, resources can be requested generically by specifying a \$ in place of the identifier, for example CPU-\$ or TAPE-\$. The \$ indicates to CONTROL-M that it must select a specific resource, that is, a resource with an identifier, to replace the generic resource, that is, the resource with the \$.

#### – NOTE



If you use the \$ to request generic resources, the \$ must appear at the end of the resource name.

If a \$ is specified for all required resource identifiers, the CONTROL-M monitor does not assign the resources unless it can assign all resources with the same identifier, for example, all resources with identifier A or all resources with identifier B.

When using the generic \$ identifier, you can use one of the following methods to ensure a specific CPU is used for processing the job:

- Use JCL AutoEdit System variable %%\$SIGN to extract the CPU identifier assigned by the CONTROL-M monitor and then to assign the job to that same CPU. System variable %%\$SIGN is discussed in Chapter 5, "JCL and AutoEdit Facility."
- Use CONTROL-M submit Exit CTMX002.

CONTROL-M Exit CTMX004 can also be used to help prevent bottlenecks caused by resource contention.

For more information on CONTROL-M exits CTMX002 and CTMX004, see the *INCONTROL for z/OS Administrator Guide*.

#### **Example 1A**

There are 12 tape drives in the data center connected to a single computer. Two tape drives must always remain free for emergencies. Therefore, only 10 drives can be used for production. The defined available quantity is set as follows: TAPE 0010.

Any user (job) wanting to use tape drives must specify the number of tapes required in the job parameters.

Figure 276 RESOURCE Parameter – Example 1A

```
JOB: EBDINPUT LIB CTM.PROD.SCHEDULE
                                                   TABLE: EBDPROD
COMMAND ===>
                                                   SCROLL===> CRSR
+-----
                                              SYSTEM ID
 SCHENV
                                                 NJE NODE
 SFT VAR
 CTB STEP AT NAME
 DOCMEM EBDINPUT DOCLIB CTM.PROD.DOC
 DAYS
                                                        AND/OR
 WDAYS
                                                    WCAI
 MONTHS 1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- Y 11- Y 12-Y
 DATES
 CONFCAL SHIFT
                       RETRO Y MAXWAIT 04 D-CAT
 DEFINITION ACTIVE FROM
                           UNTIL
 ΙN
 CONTROL
 PIPE CTM.PROD.PIPE
FROM TIME
 FROM TIME + DAYS UNTIL TIME + DAYS
DUE OUT TIME + DAYS PRIORITY SAC CONFIRM
                           PRIORITY SAC CONFIRM
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
                                                         11.17.00
```

CONTROL-M checks if there are two tape drives available. If there are, the tape drives are "given" to the job. The total number of free tapes is now eight. When the job finishes executing, the tape drives are returned to the general pool.

Suppose that many jobs are using tapes, and the available quantity is only one. A job that requires two tape drives must wait. The job is not submitted until the required number of tapes are available.

An authorized person decides that only one tape unit is needed for emergencies and adds one tape unit to the global quantity available for use. Now the maximum number of tape drives is eleven, and the number of available tape drives is two. The job is submitted.

#### **Example 1B**

The data center discussed in the previous example is expanding. It now has two computers and 20 tape drives. The tape drive distribution is:

- CPU1 only 8
- CPU2 only -8
- Transferables 6

Currently, CPU1 is connected to four transferable drives, one transferable drive is connected to CPU2, and one transferable drive is out of order. The situation is presented to CONTROL-M as follows:

```
TAPE1 12
TAPE2 7
```

A job requests three tape drives, on any computer.

Figure 277 RESOURCE Parameter – Example 1B

```
JOB: EBDINPUT LIB CTM.PROD.SCHEDULE
                                                                 TABLE: EBDPROD
COMMAND ===>
                                                                SCROLL===> CRSR
  SCHENV
                      SYSTEM ID
                                                            NJE NODE
  SET VAR
  CTB STEP AT NAME TYPE
DOCMEM EBDINPUT DOCLIB CTM.PROD.DOC
  DAYS
                                                                DCAL
                                                                     AND/OR
                                                                 WCAL
  MONTHS 1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- Y 11- Y 12-Y
  DATES
  CONFCAL
                 SHIFT RETRO Y MAXWAIT 04 D-CAT
                  PDS
  DEFINITION ACTIVE FROM
                                UNTIL
  CONTROL
  PIPE CTM.PROD.PIPE
FROM TIME
  FROM TIME + DAYS UNTIL TIME
DUE OUT TIME + DAYS PRIORITY
                                  UNTIL TIME + DAYS
PRIORITY SAC CONFIRM
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
                                                                        11.17.00
```

The result is that the available quantity of either TAPE1 or TAPE2 is reduced by three.

The CONTROL-M scheduling algorithm makes the optimal decision as to which of the two computers to send the job. It is possible to intervene in this selection process. For more information, see user Exit CTMX004 in the *INCONTROL for z/OS Administrator Guide*.

#### **Example 1C**

A job requests three tape drives on CPU1.

Figure 278 RESOURCE Parameter – Example 1C

```
JOB: FBDINPUT LIB CTM.PROD.SCHEDULE
                                                          TABLE: EBDPROD
                                                          SCROLL===> CRSR
COMMAND ===>
  SCHENV
                             SYSTEM ID
                                                       NJE NODE
  SET VAR
  SET VAR
CTB STEP AT NAME
  DOCMEM EBDINPUT DOCLIB CTM.PROD.DOC
  DAYS
                                                           DCAL
                                                               AND/OR
  WDAYS
  MONTHS 1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- Y 11- Y 12-Y
  DATES
                 SHIFT RETRO Y MAXWAIT 04 D-CAT
  CONFCAL
  MINIMUM
                 PDS
  DEFINITION ACTIVE FROM
                               UNTIL
  ΙN
  CONTROL
  RESOURCE TAPE1
                           0003
  PIPE CTM.PROD.PIPE
  FROM TIME + DAYS UNTIL TIME +
DUE OUT TIME + DAYS PRIORITY SAC
                              UNTIL TIME + DAYS
                                                 CONFIRM
                                                                11.17.00
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
```

The result is that the available quantity of the resource TAPE1 is reduced by three.

The tape drive that was out of order has been fixed. An operator makes it available for use by jobs running on CPU2 by correcting the global available quantities to:

```
TAPE2 8
```

The shift manager decides to assign two tapes from CPU1 to CPU2. The new situation as seen by CONTROL-M:

```
TAPE1 10
TAPE2 10
```

# RETENTION: # OF DAYS TO KEEP: Post— Processing Parameter

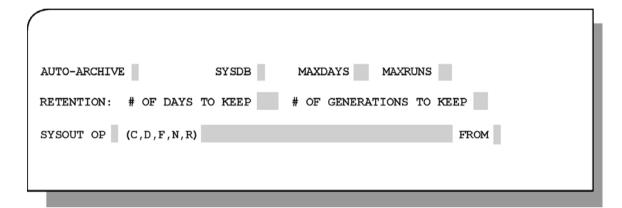
Number of days to retain the job in the History Jobs file.



#### NOTE -

At sites that do not use the History Jobs file, this parameter is not relevant and is not displayed.

Figure 279 RETENTION: # OF DAYS TO KEEP Parameter Format



Optional. Valid values are described in Table 204.

Table 204 RETENTION: # OF DAYS TO KEEP Parameter Values

Value	Description
0 through 999	Retain the job for the specified number of days. Default: 0.
''(Blank)	Retain the job according to the RETENTION: # OF GENERATIONS TO KEEP parameter.

### **General Information**

Jobs in the History Jobs file are easier to restore to the Active Jobs file, for example, for restart, than jobs archived to CDAM. Therefore, it may be desirable to retain a job in the History Jobs file for a period of time.

# OF DAYS TO KEEP enables specification of a fixed number of days to keep the job in the History Jobs file. Once the specified number of days has been reached, the job is automatically deleted from the History Jobs file during the next New Day processing.

# OF DAYS TO KEEP and # OF GENERATIONS TO KEEP are mutually exclusive. A value can be specified for either, but not both.

#### - NOTE -



When changing job criteria from retention-days to retention-generation (or vice-versa), previous job criteria are lost and are not acted upon.

For retention criteria to hold across job executions, the jobs must be identical in all respects. For example, if a job is transferred to a different group, it is treated as a different job for purposes of retention. In this case, retention values are reset, and retention is calculated from the moment of transfer.

# **Example**

Retain the archived job in the History Jobs file for 30 days.

Figure 280 RETENTION: # OF DAYS TO KEEP Parameter Example

JOB: PRDKPLO2 LIB COMMAND ===>						TABLE: PRODKPL SCROLL===> CRSR
RESOURCE PIPE TIME: FROM TIME ZONE:						
OUT AUTO-ARCHIVE Y RETENTION: # OF SYSOUT OP (C,D	DAYS TO KE			ATIONS	TO KEEP	FROM
MAXRERUN RER STEP RANGE		PROC)				FROM
ON PGMST ANYSTEP	PROCST		S S***			**** A/O
DO IFRERUN F DO RERUN DO	ROM \$ABEND					CONFIRM Y
ON PGMST DO	PROCST	CODE	S			A/0
SHOUT WHEN MS	TIME	+	DAYS	ТО		URGN
COMMANDS: EDIT, D	OC, PLAN, J	OBSTAT				07.06.04

# RETENTION: # OF GENERATIONS TO KEEP: Post—Processing Parameter

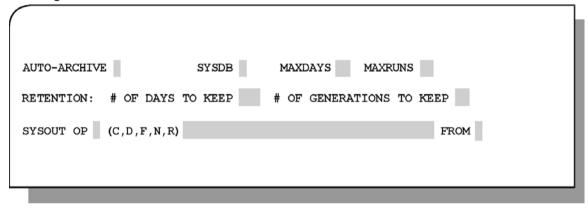
Maximum number of generations of the job to keep in the History Jobs file.



#### NOTE -

At sites that do not use the History Jobs file, this parameter is not relevant and is not displayed.

Figure 281 RETENTION: # OF GENERATIONS TO KEEP Parameter Format



Optional. Valid values are described in Table 205.

Table 205 RETENTION: # OF GENERATIONS TO KEEP Values

Value	Description
0 through 99	Retain the specified number of generations of the job.
''(Blank)	Retain the job according to the RETENTION: # OF DAYS TO KEEP parameter.

The default is 0.

### **General Information**

Jobs in the History Jobs file are easier to restore to the Active Jobs file, for example, for restart, than jobs archived to CDAM. Therefore, it may be desirable to retain several of the most current generations of the job in the History Jobs file.

# OF GENERATIONS TO KEEP enables specification of the number of generations of the job to keep in the History Jobs file. Once the specified number of generations has been reached, as a new generation is added to the History Jobs file, the earliest remaining generation is deleted.

# OF DAYS TO KEEP and # OF GENERATIONS TO KEEP are mutually exclusive. A value can be specified for either, but not both.

#### - NOTE -



When changing job criteria from retention-days to retention-generation, or vice versa, previous job criteria are lost and are not acted upon.

For retention criteria to hold across job executions, the jobs must be identical in all respects. For example, if a job is transferred to a different group, it is treated as a different job for purposes of retention. In this case, retention values are reset, and retention is calculated from the moment of transfer.

# **Example**

Retain up to 10 generations of the archived job in the History Jobs file.

Figure 282 RETENTION: # OF GENERATIONS TO KEEP Parameter Example

OB: PRDKPLO2 LIB	CTM.PROD.SCH					TABLE: PRODKPL SCROLL===> CRSR
RESOURCE PIPE TIME: FROM TIME ZONE:					SAC	CONFIRM
OUT AUTO-ARCHIVE Y RETENTION: # 0 SYSOUT OP (C,I MAXRERUN RE STEP RANGE	F DAYS TO KEE D,F,N,R) RUNMEM	P #	OF GENER	ATIONS	TO KEEP 1	O FROM
ON PGMST ANYSTE		COD				***** A/0
DO IFRERUN DO RERUN DO	FROM \$ABEND		ТО			CONFIRM Y
ON PGMST DO	PROCST	COD	ES			A/0
SHOUT WHEN MS	TIME	+	DAYS	T0		URGN
COMMANDS: EDIT,	DOC, PLAN, JO	BSTAT				07.6.04

# **RETRO: Basic Scheduling Parameter**

Enables the job to be scheduled after its original scheduling date has passed.

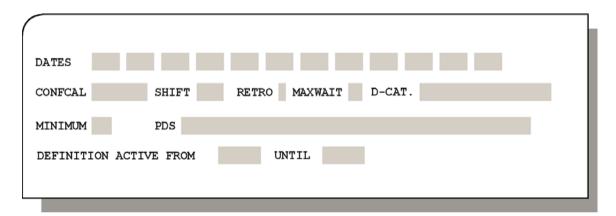
#### — WARNING



BMC does not recommend the use of the RETRO parameter, which is documented here for backward compatibility reasons only. For workload scheduling after the original scheduling date has passed, use enhanced CONTROL-M Newday functionality. For more information, see the *INCONTROL for z/OS Administrator Guide*, "CONTROL-M," "Special Newday Parameters."

The RETRO=Y parameter should be removed from any job scheduling definitions.

Figure 283 RETRO Parameter Format



Optional. Valid values are described in Table 206.

Table 206 RETRO Values

Value	Description
Y (Yes)	Allow scheduling of the job after its original scheduling date has passed
N (No)	Do not allow scheduling of the job after its original scheduling date has passed. Default.

## **General Information**

The RETRO parameter is used to control situations where the computer has not been working for a day or more due to holiday, hardware failure, and so on.

When such situations occur, it is necessary to instruct CONTROL-M whether the job is to be retroactively scheduled for the days when the computer (or CONTROL-M) was inactive.

- When Y is specified for the RETRO parameter, job orders are placed in the Active Jobs file for all the days the job ought to have been originally scheduled. Scheduling occurs from the last scheduling date to the current working date, provided that those days were included in one of the Basic Scheduling parameters (DAYS, DCAL, and so on). Each job order placed on the Active Jobs file is associated with a different original scheduling date. For additional information see "MAXWAIT: Basic Scheduling Parameter" on page 519.
- When N is specified for the RETRO parameter, the job is scheduled only if the current working date is a date on which the job is normally scheduled.

The RETRO parameter cannot be used with the MINIMUM and PDS parameters, nor in group scheduled jobs; if specified in Group scheduled jobs, the parameter is ignored.

# **Examples**

#### **Example 1**

Schedule the job only on specified days of the month. If the date has passed, do not schedule the job.

Figure 284 RETRO Parameter – Example 1

```
TABLE: PRODKPL
JOB: PRDKPL01 LIB CTM.PROD.SCHEDULE
COMMAND ===>
                                                                  SCROLL===> CRSR
  MEMNAME PRDKPLO1 MEMLIB CTM.PROD.JCL
OWNER SYS1 TASKTYPE JOB PREVENT-NCT2 DFLT N
APPL KPL GROUP PROD-KPL
  APPL KPL GROUP PROD-KPL
DESC DAILY PRODUCTION - START OF APPL-PROD-KPL
  OVERLIB
                                                               STAT CAL
                                 SYSTEM ID
  SCHENV
                                                               NJE NODE
  SET VAR
  DOCMEM PRDKPL01 DOCLIB CTM.PROD.DOC
  DAYS 15,16,18,19,20
                                                                    DCAL
                                                                         AND/OR
  WDAYS
  MONTHS 1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- Y 11- Y 12- Y
  DATES
                    SHIFT RETRO N MAXWAIT 00 D-CAT
  CONFCAL
  MINIMIM
  DEFINITION ACTIVE FROM
                                  UNTIL
  ΤN
  CONTROL
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
                                                                11.17.00
```

Assume the computer was offline from the 16th up to and including the 18th, and the 15th was the last date that the job was scheduled for execution. Today is the 19th. Therefore, the job is only scheduled for execution on the 19th.

### **Example 2**

Schedule the job for all working days, even if the computer is not active:

DCAL WORKDAYS RETRO Y

Assume the WORKDAYS calendar contains the dates 15, 16, 18, and 19, and the same conditions as above exist. The job is scheduled three times with the original scheduling dates: the 16th, the 18th and the 19th.

# **SAC: Run Time Parameter**

This parameter is specifically designed for users who have converted from third-party vendor job scheduling products. BMC Software does not recommend its use otherwise. It enables all scheduled runs of the job to be shifted to their proper scheduling days.

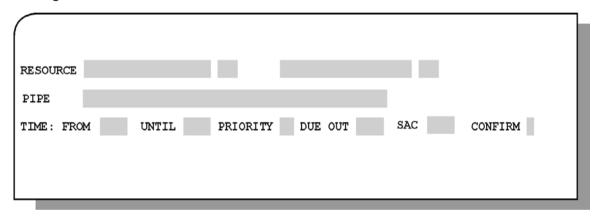
#### - NOTE



Do not use the SAC parameter unless specifically required in conjunction with the conversion. Check the Conversion Guide for your specific application.

A related parameter is discussed in "DESC: General Job Parameter" on page 432.

Figure 285 SAC Parameter Format



Optional. Valid values are shown in Table 207.

**Table 207 SAC Parameter Values** 

Value	Description
P (Previous)	Changes the scheduling of the job to the previous day.
N (Next)	Changes the scheduling of the job to the next day.
- (Group Previous)	For jobs in Group tables only. Changes the scheduling of the job to the previous day.
+ (Group Next)	For jobs in a Group table only. Changes the scheduling of the job to the next day.
''(Blank)	No changes need be made. Default.

### **General Information**

Many scheduler products do not allow the site to define the time of the new working day. Instead, those products fix the time, such as midnight. CONTROL-M, however, allows the site to define when the new working day starts.

In most cases, this added CONTROL-M flexibility can be utilized without additional adjustment. Occasionally, however, an adjustment to the job schedule may be required due to the differences between the start of the working day. The SAC parameter is used to perform such an adjustment.

For information on the correct usage of the SAC parameter, see the Conversion Guide provided for your specific product.

#### - NOTE



Unless you are certain that SAC must be used, and you are certain how to use it, leave this parameter blank.

# **Example**

Due to differences in the time of the start of the new working day, shift the scheduling of the following converted job back to the previous day.

Figure 286 SAC Parameter Example

```
TABLE: PRODKPL
JOB: CAPRKL1 LIB CTM.PROD.SCHEDULE
COMMAND ===>
                                                           SCROLL===> CRSR
  DOCMEM IEFBR14 DOCLIB CTMP.DOC
  DAYS ALL
                                                             DCAL
                                                                 AND/OR
                                                            WC.AI
  MONTHS 1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- Y 11- Y 12- Y
  DATES
  CONFCAL
                 SHIFT
                             RETRO N MAXWAIT 04 D-CAT
                  PDS
  DEFINITION ACTIVE FROM
                               UNTIL
                              Ε
  CONTROL MK-1
  RESOURCE
  PIPF
 FROM TIME + DAYS
DUE OUT TIME + DAYS
  FROM TIME
                                UNTIL TIME
                                           SAC P CONFIRM
                                PRIORITY
  TIME ZONE:
  OUT
          COND1
                              ODAT + COND2
                                                           ODAT -
                                       COND4
          COND3
                              ODAT +
                                                           ODAT -
                                                                  15.24.05
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
```

# **SCHEDULE TAG: Basic Scheduling Parameter**

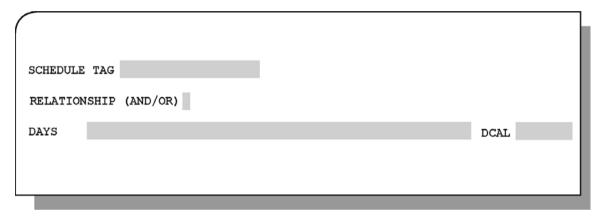
Identifies a set of scheduling criteria for Group scheduling. This parameter appears only in Group scheduling tables.

Mandatory for Group Entities. Any alphanumeric value of 1 to 20 characters.

Optional for job scheduling definitions. Must be either a schedule tag value defined in the Group Entity, or the value *.

A related parameter is RELATIONSHIP, which is described on page 591.

Figure 287 SCHEDULE TAG Parameter Format



Upon specifying a schedule tag value and pressing **Enter**, a new SCHEDULE TAG field is opened, for specifying additional schedule tags. In Group Entities, a new set of basic scheduling parameters is opened with the new SCHEDULE TAG field to allow specification of criteria to be associated with the new tag.

### **General Information**

A Group Entity contains sets of basic scheduling criteria to be applied to job scheduling definitions in the Group scheduling table. Each set of basic scheduling criteria in the Group Entity is assigned a unique label, specified in the SCHEDULE TAG field, which is used for referencing that set of criteria.

At least one schedule tag, with basic scheduling criteria, must be defined in the Group Entity.

To apply any sets of Group Entity basic scheduling criteria to a job scheduling definition, specify the schedule tag names of the desired criteria in the SCHEDULE TAG fields in the job scheduling definition.

If multiple SCHEDULE TAG values are specified in the job scheduling definition, tags are checked sequentially during job scheduling to determine if the criteria are satisfied. Once a set of schedule tag criteria are satisfied, no other schedule tags in the job are checked.

An asterisk (*) can be specified as a SCHEDULE TAG value in the job scheduling definition. When checks are performed for a schedule tag with a value of *, the Group Entity's scheduling tags are searched in reverse order to determine which set of basic scheduling criteria are satisfied on the particular day, and the chosen tag is applied to the job.

Each job scheduling definition can have its own basic scheduling criteria defined, independent of the schedule tag criteria in the Group Entity.

Jobs in a Group scheduling table are eligible for scheduling on a particular day only if at least one set of basic scheduling criteria in the Group Entity are satisfied.

If a group is eligible for scheduling on a particular day, a job in the group is scheduled in any of the following cases:

- The value in the RELATIONSHIP parameter is O (OR), and either the basic scheduling criteria of the job or a set of its schedule tag criteria (or both) are satisfied.
- The value in the RELATIONSHIP parameter is A (AND), and its basic scheduling criteria and a set of its schedule tag criteria are both satisfied.

When the name of a SCHEDULE TAG definition is modified or deleted, a confirmation panel is displayed. If "Y" is entered, the changed or deleted name is propagated throughout all of the group's job definitions, that is, all the corresponding schedule tag references in the job definition are changed or deleted. If "N" is entered, no changes are made.

### **Examples**

#### Example 1

For a Group Entity:

The Group Entity for group ACCOUNTS_GROUP in Group scheduling table ACCOUNTS contains two sets of basic scheduling parameters. One set is identified by schedule tag ALL_DAYS, and the other set is identified by schedule tag SUNDAYS.

#### Figure 288 SCHEDULE TAG Parameter - Example 1

```
GRP ACCOUNTS_GROUP CTM.PROD.SCHEDULE(GRP)
COMMAND ===>
                                                          SCROLL===> CRSR
  GROUP ACCOUNTS_GROUP MEMNAME ACCOUNTS
  OWNER NO4B
  APPL
  DESC
  ADJUST CONDITIONS N GRP MAXWAIT
  SFT VAR
  DOCMEM ACCOUNTS DOCLIB CTM.PROD.DOC
  SCHEDULE TAG ALL_DAYS
  DAYS ALL
                                                              AND/OR
                                                          WCAL
  MONTHS 1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- Y 11- Y 12- Y
  DATES
  CONFCAL SHIFT RETRO N MAXWAIT 00 D-CAT SCHEDULE TAG ACTIVE FROM UNTIL
  SCHEDULE TAG SUNDAYS
                                                          DCAL
                                                              AND/OR
  WDAYS 01
                                                          WCAL
  MONTHS 1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- Y 11- Y 12- Y
  SCHEDULE TAG ACTIVE FROM UNTIL
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
                                                                18.19.14
```

#### **Example 2**

For a job scheduling definition:

Schedule job TABHOURS when the basic scheduling criteria identified by schedule tag ALL_DAYS in the Group Entity (in Example 1A) are satisfied.

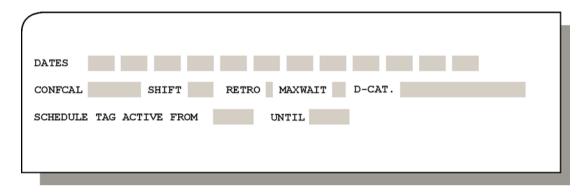
Figure 289 SCHEDULE TAG Parameter – Example 2

```
JOB: TABHOURS LIB CTM.PROD.SCHEDULE
                                                        TABLE: ACCOUNTS
COMMAND ===>
                                                         SCROLL===> CRSR
  MEMNAME TABHOURS MEMLIB CTM.PROD.JCL
  OWNER NO4B TASKTYPE JOB PREVENT-NCT2 DFLT N
                                   GROUP ACCOUNT_GROUP
  APPL
  DESC TABULATE EMPLOYEE HOURS
  OVERLIB
                                                        STAT CAL
  SCHENV
                              SYSTEM ID
                                                        NJE NODE
  SET VAR
  CTB STEP AT
                    NAME
                                   TYPE
  DOCMEM TABHOURS DOCLIB CTM.PROD.DOC
  SCHEDULE TAG ALL_DAYS
  SCHEDULE TAG
  RELATONSHIP (AND/OR) 0
                                                            DCAL
  DAYS
                                                                AND/OR
  WDAYS
  MONTHS 1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- Y 11- Y 12- Y
  DATES
  CONFCAL
                  SHIFT RETRO N MAXWAIT OO D-CAT
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
                                                                 18.36.07
```

# SCHEDULE TAG ACTIVE: Basic Scheduling Parameter

Specifies the time limits (FROM, UNTIL) for using the schedule tag.

Figure 290 SCHEDULE TAG ACTIVE Parameter Format



Optional. The parameter includes the subparameters described in Table 208.

**Table 208 SCHEDULE TAG ACTIVE Subparameters** 

Subparameter	Description
FROM	6-digit date. A job that refers to this schedule tag will only be ordered if the ordering date is later than the date specified. Default: ' '(Blank)
UNTIL	6-digit date. A job that refers to this schedule tag will only be ordered if the ordering date is earlier than the date specified. Default: ' '(Blank)

The format of either the FROM or UNTIL subparameters may be ddmmyy, mmddyy, or yymmdd, depending on your local site standard, as set by the DATETYP parameter in the IOAPARM member in the IOA PARM library.

# **General Information**

The FROM and UNTIL dates together define a time frame for ordering jobs that have a specific schedule tag within the Group that defines that schedule tag.

The following cases apply:

■ If you specify both the FROM and UNTIL subparameters for a particular schedule tag, jobs within the group that refer to that schedule tag can only be ordered on or later than the date specified in the FROM subparameter, and on or earlier than the date specified in the UNTIL subparameter. There are two possibilities:

1. The date specified in the FROM subparameter is earlier than that specified in the UNTIL subparameter.

For example,

```
SCHEDULE TAG ACTIVE FROM 091001 UNTIL 011101
```

Jobs within the group that refer to this schedule tag can only be ordered on or between October 9, 2001 and November 1, 2001.

2. The date specified in the FROM subparameter is later than that specified in the UNTIL subparameter.

For example,

```
SCHEDULE TAG ACTIVE FROM 090501 UNTIL 010401
```

Jobs within the group that refer to this schedule tag can only be ordered on or after May 9, 2001, or before or on April 1, 2001, but not between those dates.

■ If you specify the FROM subparameter for a particular schedule tag, but not the UNTIL subparameter, jobs within the group that refer to that schedule tag cannot be ordered before the date specified, but can be ordered on that date or any date later than that date.

For example,

```
SCHEDULE TAG ACTIVE FROM 091001 UNTIL
```

Jobs within the group that refer to this schedule tag can only be ordered on or after October 9, 2001.

■ If you do not specify the FROM subparameter for a particular schedule tag, but specify the UNTIL subparameter, jobs within the group that refer to the same schedule tag cannot be ordered after the date specified, but can be ordered on that date or any date earlier than that date.

For example,

```
SCHEDULE TAG ACTIVE FROM UNTIL 011101
```

Jobs within the group that refer to this schedule tag can only be ordered before or on November 1, 2001.

■ If you do not specify either the FROM or UNTIL subparameters, there is no restriction on the date when jobs within the group can be ordered.

For example,

SCHEDULE TAG ACTIVE FROM

UNTIL

Jobs within the group that refer to this schedule tag can be ordered on any date.

■ If a job specifies more than one schedule tag and one of the Schedule Tag definitions is such that the job can be ordered on a particular day, the job will be ordered even if it would not be ordered under the terms of another of its schedule tag definitions.

For example, if within a Group Entity one schedule tag is specified as

SCHEDULE TAG ACTIVE FROM 0

091001

UNTIL

011101

and another schedule tag is specified as

SCHEDULE TAG ACTIVE FROM

UNTIL

jobs within the Group that have both these schedule tags can be ordered on any date.

# **Example**

Schedule Tag A schedules jobs to run on the 5th of each month. Job B should be scheduled according to the criteria of Schedule Tag A; however, it should only run until January 15, 2010.

In the group definition:

SCHEDULE TAG A DAYS 05

#### In job definition B:

SCHEDULE TAG A RELATIONSHIP (AND/OR) A DAYS ALL

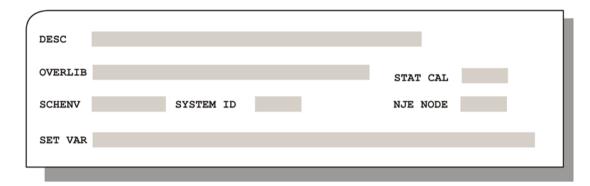
DEFINITION ACTIVE FROM

UNTIL 100115

# **SCHENV: General Job Parameter**

Name of the workload management scheduling environment that is to be associated with the job.

Figure 291 SCHENV Parameter Format



SCHENV specifies a scheduling environment of 1 through 16 characters. Only trailing blanks are allowed.

By default, the SCHENV parameter is optional.

#### **General Information**

If a value is specified for the SCHENV parameter, the JCL job statement is modified by the addition of a statement in the following form:

// SCHENV=schedule_environment

If a value is specified for the SCHENV parameter, it will not override any scheduling environment specified in the job statement unless the OVERJCLM parameter in the CTMPARM library is set to Y.

If a value is specified for the SCHENV parameter, before job submission, CONTROL-M checks whether the specified WLM scheduling environment is available before actually submitting the job. If the scheduling environment is not available, CONTROL-M places the job in WAIT SCHEDULE status and waits until the WLM scheduling environment is available, before submitting the job.

If the task type is a started task, SCHENV is protected. If the task type is changed from a job to a started task, SCHENV is erased and protected.

# **Example**

If the scheduling environment of job ACCT01 is to be SCHD2, specify the following:

DESC				
OVERLIB			STAT CAL	
SCHENV	SCHD2	SYSTEM ID	NJE NODE	

The job statement is modified as follows:

```
//ACCT01 JOB ,PROD1,CLASS=A,MSGCLASS=X,
// MSGLEVEL=(1,1),
// SCHENV=SCHD2
```

# **SET VAR: General Job Parameter**

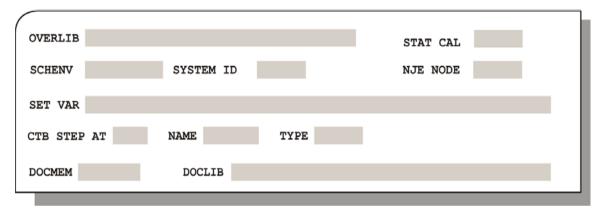
Assigns a value to an AutoEdit variable that can be used to set up the JCL of the job or to define a Global variable in the IOA Global Variable Database.



#### NOTE -

SET VAR and DO SET statements are similar, but not identical. The differences are outlined below in "Differences between SET VAR and DO SET" on page 623.

Figure 292 SET VAR Parameter Format



Optional. Valid values must comply with the following requirements:

■ They must not contain blanks. If you want to specify an embedded blanks in a SET VAR expression, use the %%BLANK*n* AutoEdit variable.

For example,

%%A=TODAY%%BLANK1%%.IS%%BLANK1%%.SUNDAY

resolves to

**TODAY IS SUNDAY** 

For more information see "Non-Date System Variables" on page 748.

■ They must be specified in the format

%%variable=expression

#### In this format:

- %%variable is a user-defined AutoEdit variable
- *expression* is any of the following components, provided it resolves to a single value:
  - a value (for example, 5)
  - an AutoEdit System variable or previously defined user-defined variable, for example, %%0DATE
  - a compound expression that contains constants, AutoEdit variables, or AutoEdit functions, or any combination of these, for example %%\$CALCDTE.%%BLANK.%%ODATE.%%BLANK%%.+5

#### - WARNING -



AutoEdit function %%SUBSTR should not be used in the expression portion of a SET VAR variable definition. Doing so may cause errors when trying to resolve the variable's value.

### **General Information**

A major advantage of using AutoEdit variables is that the JCL can be submitted with different values for different executions without actually changing the JCL.

There are two types of AutoEdit variables:

- System variables that are assigned values by the system
- user-defined variables for which the user must supply values.
   These variables can be either local or global.

One method of supplying a value for a user-defined variable is by defining the variable and its value in a SET VAR statement. The value is assigned at time of job submission.

At the time of job submission, AutoEdit variables in the JCL are resolved in sequence. By default, if an AutoEdit variable cannot be resolved, the job is not submitted. This default can be changed using an appropriate %%RESOLVE AutoEdit control statement.

SET VAR statements can also be used to define and update Global Variables in the IOA Global Variable Database. For more information on Global Variables, including Global Variable syntax, see "Global Variables" on page 760.

As of version 6.0.00, SET VAR variables defined in a Group entity are available to all the jobs in the group. However, they do not override SET VAR variables defined in the job scheduling definition.

An unlimited number of SET VAR statements can be specified.

Upon filling in a SET VAR statement and pressing **Enter**, a new blank SET VAR statement is displayed.

JCL Setup and the AutoEdit facility are described in depth in Chapter 5, "JCL and AutoEdit Facility."

#### **Differences between SET VAR and DO SET**

SET VAR and DO SET statements are similar but have the following differences:

- Local variables in SET VAR statements are always applied before the job is submitted. DO SET is a post-processing statement that can only be applied after its accompanying ON step and code criteria are satisfied. This means that a local value specified in the DO SET statement can only be applied in the next submission of the job (that is, for cyclic and rerun or restarted jobs).
- Global variables specified in a SET VAR statement are defined or updated in the IOA Global Variable database before job submission. Global variables specified in a DO SET statement are defined or updated in the IOA Global Variable database as part of job post-processing.
- A SET VAR statement appears in each job scheduling definition. A DO SET statement does not appear unless specified. To specify a DO SET statement, type SET in an empty DO field and press **Enter**.
- In a SET VAR statement, the parameter value is specified after the keyword VAR. In a DO SET statement, the parameter value is specified after the keyword VAR.

### **Examples**

#### **Example 1**

In this example, AutoEdit statements in the job scheduling definition and the JCL allocate space for the job. If the job abends due to insufficient space, the AutoEdit statements adjust the allocated space and rerun or restart the job.

The following step in the JCL of the job sets the quantity of available space to five units of whatever type (track or cylinder) is specified in the job scheduling definition.

```
//STEP10 EXEC PGM=MYPGM
//OUTFILE DD DSN=NEWFILE,DISP=(NEW,CATLG,DELETE),
// SPACE=(%%SPACE_TYPE,5),UNIT=SYSDA
```

The job scheduling definition contains the following SET VAR statement that sets the space type to "track":

```
SET VAR %%SPACE_TYPE=TRK
```

In this case, the second line in the above DD statement resolves to:

```
// SPACE=(TRK,5),UNIT=SYSDA
```

The job scheduling definition also contains the following statements that are activated if the job abends due of lack of space (code S*37). These statements change the space type to "cylinder", which provides enough space, and rerun the job. If CONTROL-M/Restart is active, the job is restarted from the abended step.

```
ON PGMST STEP10 CODES S*37

DO SET %%SPACE_TYPE = CYL

[DO IFRERUN FROM $ABEND] ===> If CONTROL-R is active

DO RERUN
```

If the job abends due to insufficient space, the second line of the earlier JCL DD statement resolves to:

```
// SPACE=(CYL,5),UNIT=SYSDA
```

when the job is submitted for rerun (or restart).

#### **Examples 2A and 2B**

The following examples show how one job scheduling definition and one JCL member can be used for both the test environment and the production environment by changing the value of only one parameter, the SET VAR parameter.

Assume the following JCL for the job:

```
//PRDKPL01 JOB 0,M22,CLASS=A,MSGCLASS=X,REGION=4000K
//STEP01 EXEC %%PROC%%.INPT
//STEP02 EXEC %%PROC%%.UPDT
//STEP03 EXEC %%PROC%%.RPTS
```

#### **Example 2A**

The following job scheduling definition replaces the %%PROC variable in the EXEC statements of the JCL with procedure name prefix TEST.

Figure 293 SET VAR Parameter Example - 2A

```
JOB: PRDKPL01 LIB CTM.PROD.SCHEDULE
                                                             TABLE: PRODKPL
COMMAND ===>
                                                             SCROLL===> CRSR
  MEMNAME PRDKPL01 MEMLIB CTM.PROD.JCL
  OWNER SYS1 TASKTYPE JOB PREVENT-NCT2 DFLT N
                                    GROUP PROD-KPL
  APPL KPL
  DESC
         DAILY PRODUCTION - START OF APPL-PROD-KPL
  OVERLIB
                                                         STAT CAL
                                                         NJE NODE
  SCHENV
                               SYSTEM ID
  SET VAR %%PROC=TEST
  SET VAR
  CTB STEP AT
                    NAME
                                    TYPE
  DOCMEM PRDKPL01 DOCLIB CTM.PROD.DOC
  SCHEDULE TAG
  RELATIONSHIP (AND/OR)
  DAYS
         01
                                                             DCAL
                                                                 AND/OR
                                                             WCAL
  WDAYS
  MONTHS 1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- Y 11- Y 12- Y
  DATES
  CONFCAL
                  SHIFT
                             RETRO Y MAXWAIT OO D-CAT
                  PDS
  MINIMUM
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
                                                                   18.36.07
```

When a SET VAR statement is used to specify %%PROC=TEST, the JCL is resolved as follows:

```
//PRDKPL01 JOB 0,M22,CLASS=A,MSGCLASS=X,REGION=4000K
//STEP01 EXEC TESTINPT
//STEP02 EXEC TESTUPDT
//STEP03 EXEC TESTRPTS
```

#### **Example 2B**

The job scheduling definition has now been modified to replace the procedures (%%PROC) used in the job with production (PROD) procedures.

Figure 294 SET VAR Parameter Example 2B

```
JOB: PRDKPL01 LIB CTM.PROD.SCHEDULE
                                                           TABLE: PRODKPL
COMMAND ===>
                                                           SCROLL===> CRSR
  MEMNAME PRDKPL01 MEMLIB CTM.PROD.JCL
  OWNER SYS1 TASKTYPE JOB PREVENT-NCT2 DFLT N
                                   GROUP PROD-KPL
  APPL KPL
        DAILY PRODUCTION - START OF APPL-PROD-KPL
  DESC
  OVERLIB
                                                        STAT CAL
                                                        NJE NODE
  SCHENV
                              SYSTEM ID
  SET VAR %%PROC=PROD
  SET VAR
  CTB STEP AT
                    NAME
                                   TYPE
  DOCMEM PRDKPL01 DOCLIB CTM.PROD.DOC
  SCHEDULE TAG
  RELATIONSHIP (AND/OR)
  DAYS
         01
                                                            DCAL
                                                                AND/OR
                                                            WCAL
  WDAYS
  MONTHS 1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- Y 11- Y 12- Y
  DATES
  CONFCAL
                 SHIFT
                             RETRO Y MAXWAIT OO D-CAT
                 PDS
  MINIMUM
  DEFINITION ACTIVE FROM
                               UNTIL
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
                                                                 18.36.07
```

When a SET VAR statement is used to specify %%PROC=PROD, the JCL is resolved as following:

```
//PRDKPL01 JOB 0,M22,CLASS=A,MSGCLASS=X,REGION=4000K
//STEP01 EXEC PRODINPT
//STEP02 EXEC PRODUPDT
//STEP03 EXEC PRODRPTS
```

# **SHOUT: Post-Processing Parameter**

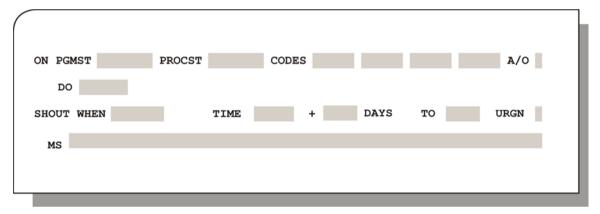
Sends ("shouts") a message to a destination when a specific situation occurs.

#### **NOTE**



DO SHOUT and SHOUT statements are similar, but not identical. The differences are outlined below in "Differences between SHOUT and DO SHOUT" on page 635.

Figure 295 SHOUT Parameter Format



Optional. Upon filling in the SHOUT statement and pressing **Enter**, a new SHOUT statement is opened.

Each SHOUT statement consists of four subparameters: WHEN (situation), TO (destination), URGN (urgency), and MS (message text).

Table 209 SHOUT Subparameters (part 1 of 4)

Subparameter	Description
WHEN	Situation in which to send the message. Valid values are:
	■ OK – The message is sent if the job ends OK.
	■ NOTOK – The message is sent if the job ends NOTOK.
	■ RERUN – The message is sent if the job is rerun and DO RERUN is specified in ON PGMST.
	■ LATESUB <i>time + days</i> – The message is sent if the job was not submitted by the specified day and time offset, where:
	<ul> <li>If Days is entered, it must be a number between zero and 120.</li> <li>Time is in the format hhmm or an *. If * is specified, the CONTROL-M monitor uses the calculated DUE IN date and time of the job, as displayed in the Zoom screen, to determine if the job was not submitted at the correct date and time. For more information, see "Automatic Job Flow Adjustment" on page 74.</li> <li>Note that if the time is *, Days must be blank.</li> </ul>
	■ LATE <i>time</i> + days – The message is sent if the job was submitted but did not finish executing by the specified day and time offset, where:
	<ul> <li>If Days is entered, it must be a number between zero and 120.</li> <li>Time is in the format hhmm or an *. If * is specified, the CONTROL-M monitor uses the calculated DUE IN date and time of the job, as displayed in the Zoom screen, to determine if the job was late. For more information, see "Automatic Job Flow Adjustment" on page 74.</li> <li>Note that if the time is *, Days must be blank.</li> </ul>

Table 209 SHOUT Subparameters (part 2 of 4)

Subparameter	Description				
	■ EXECTIME <i>limit</i> – The message is sent if the elapsed runtime of the job is outside a specified limit. The limit can be expressed as a runtime limit, or as a deviation from the average runtime of the job. Valid formats for limit are (where <i>n</i> is a 3-digit nonzero value):				
	>n – The elapsed runtime of the job is greater than n minutes. <i>n</i> cannot exceed 999.				
	<ul><li>&lt; n - The elapsed runtime of the job is less than n minutes. n cannot exceed 999.</li></ul>				
	+ <i>n</i> - The elapsed runtime of the job exceeds the average execution time of the job by at least <i>n</i> minutes. <i>n</i> cannot exceed 999.				
	<ul> <li>-n - The elapsed runtime of the job is at least n minutes less than its average execution time. n cannot exceed 999.</li> </ul>				
	+n% – The elapsed runtime of the job exceeds its average execution time by at least n%. n cannot exceed 900 –n% – The elapsed runtime of the job is at least n% less than its average execution time. n cannot exceed				
	99.  SHOUT WHEN LATE and SHOUT WHEN LATESUB are only activated for the first run of a job, not for subsequent job reruns.				
	SHOUT WHEN EXECTIME limits coded in Group entities are not continuously checked, but only when each job in the group finishes executing.				
ТО	Destination of the message (1 through 16 characters).				
	Mandatory. Valid values are:				
	<ul> <li>U-userid or USERID-userid – Writes the message to the IOA Log file under the specified user ID. userid must be 1 through 8 characters.</li> </ul>				
	■ OPER[-n] – Sends a rollable message to the operator console. n is an optional 2-digit route code. If a route code is not specified, the default routes are Master Console and Programmer Information (1 and 11), and optionally, CONTROL-M/Enterprise Manager. For more detailed information regarding route codes, refer to the IBM publication <i>Routing and Descriptor Codes, GC38-1102</i> .				
	■ OPER2[-n] – Sends an unrollable, highlighted message to the operator console. n is an optional 2 through digit route code. If a route code is not specified, the default routes are Master Console and Programmer Information (1 and 11), and optionally, CONTROL-M/Enterprise Manager. For more detailed information regarding route codes, refer to the IBM publication <i>Routing and Descriptor Codes, GC38-1102</i> .				

Table 209 SHOUT Subparameters (part 3 of 4)

Subparameter	Description
	■ [TSO - loginid   T - loginid] [;Nn   ;Mm   ;NnMm   ;Lname] – Sends the message to the specified ID (groupid or logonid). ID is mandatory.
	If a groupid is specified, it must be a valid ID found within the IOA Dynamic Destination Table.
	If a logonid is specified, it must be 1 through 7 characters.  An optional second value, indicating the computer and/or node (such as Mm) of the TSO logonid, can be specified, as follows:  Under JES2:
	Valid values are: Nn, Mm or NnMm, where:
	<ul> <li>- m is the machine ID (the computer in JES2, not the 4-character SMF system ID). For more information, see the description of specifying IOA CPUs in the discussion of the customization process in the <i>INCONTROL for</i></li> </ul>
	z/OS Installation Guide. – n is the 1 to 2 character JES/NJE node ID. Under JES3:
	The only valid value is <i>Lname</i> , where <i>Lname</i> is the logical JES name of the machine (that is, the name as used in JES3 command *T, not the SMF system ID.
	For more information, see the description of specifying IOA CPUs in the discussion of the customization process in the INCONTROL for z/OS Installation Guide.)
	<b>Note</b> : A shout to a TSO user performs a TSO SEND command, which may require authorization at the receiving node.
	■ U-M: <i>mail-name</i> – Sends a message by mail to the recipient identified by the <i>mail-name</i> prefix (1 through 12 characters).
	<ul> <li>U-S:snmp_dest - Sends an SNMP trap (message) to the recipient identified by snmp_dest.</li> <li>snmp_dest consists of from 1 through 12 characters, and can be any of the following:         <ul> <li>a host name</li> </ul> </li> </ul>
	<ul> <li>— an IP address</li> <li>— a nickname defined in the SNMPDEST destination table</li> <li>— a group name defined in the SNMPDEST destination table</li> </ul>
	<ul> <li>U-ECS – Sends messages to the CONTROL-M/Enterprise Manager user.</li> </ul>
	<b>Note:</b> If you want SHOUT Messages to be sent to the CONTROL-M/Enterprise Manager, you must install Sample Exit IOAX034W, which is in the IOA SAMPEXIT library.

Table 209 SHOUT Subparameters (part 4 of 4)

Subparameter	Description
URGN	<ul> <li>Determines the priority level of the message. Valid values are:</li> <li>■ R – Regular. Default.</li> <li>■ U – Urgent.</li> <li>■ V – Very urgent.</li> </ul>
MS	Message text. Maximum length: 70 characters.  AutoEdit variables (both system and user-defined) are supported and automatically resolved (replaced) at the time the SHOUT message is issued. For AutoEdit usage information, see Chapter 5, "JCL and AutoEdit Facility."

#### **General Information**

The message is sent to the specified destination when the WHEN condition is satisfied. The relationship between multiple SHOUT statements is OR (that as, each statement is evaluated and performed independently of the others).

AutoEdit variables (system- and/or user-defined) in the message text are supported and automatically resolved (replaced) when the SHOUT message is issued. For more information, see Chapter 5, "JCL and AutoEdit Facility."

SHOUT statements can also be defined in Group entities, where they are used in a manner similar to jobs. For example, SHOUT WHEN OK is activated when all the jobs in the group end OK.

#### The WHEN Subparameter

If SHOUT WHEN EXECTIME values are stated with a + or - sign, that is, when elapsed runtime is compared to average runtime, the shout applies only if there is a Job Statistics record for the job, containing statistics for at least one of the last 200 runs of the job.

If a Job Statistics record exists, all available elapsed-time statistics for the last 200 job runs are averaged to generate the average runtime, and the current runtime is compared to this figure according to the specified criteria.

If no Job Statistics file exists, or a record for the job does not exist, that is, there are no elapsed-time statistics for any of the last 200 job runs, the SHOUT is not activated.

#### - NOTE



Your INCONTROL administrator can tell you if the job has a Statistics file, and if the Statistics file is updated after each job run.

If EXECTIME values are negative (that is, if they are –n or –n%), the check can be performed only after the job has finished running.

When EXECTIME values are positive (that is, if they are +n or +n%), the check can be performed (and if the elapsed runtime limits are exceeded, the message can be "shouted") before the job has finished running.

When CONTROL-M calculates EXECTIME values, such as job start time, average execution time, actual elapsed time, shout message time, and so on, calculations are made only in minutes, and seconds are ignored. Therefore, the results of expressions such as SHOUT WHEN EXECTIME >001 (or +001) are unpredictable. BMC Software recommends that you use SHOUT WHEN EXECTIME only when you need to monitor jobs of more than a few minutes duration.

Relative EXECTIME limits must not exceed 24 hours. When relative EXECTIME limits exceed 24 hours (such as if +n(%) of the average runtime exceeds 24 hours), the message is "shouted" if and when processing reaches 24 hours.

If a relative EXECTIME is not specified prior to job submission, but is specified afterwards (that is, the job is held, the parameters changed in the Zoom screen, and the job is then freed), the EXECTIME value is ignored.

When the New Day procedure runs, any unexecuted SHOUT statements that relate to jobs ordered on the previous day are automatically cancelled.

If, when you order jobs, you often specify a LATE or LATESUB time that crosses the New Day time, you should consider implementing Wish WM2344. This Wish enables jobs to operate with a "shifted" New Day time for SHOUT purposes. You can find Wish WM2344 in the IOADFLT member in the IOA IOAENV library.

If you want only some specific jobs to operate with a "shifted" New Day time for SHOUT purposes, you may not want to implement Wish WM2344. An alternative method for use in such a case is illustrated in Example 4 on page 637.

#### The TO Subparameter

Specify TO=USERID-userid to write the message to the IOA Log file under the user ID specified in the parameter.

Specify TO=OPER[-n] to send the message to the operator console (route code n). If the n value is omitted, the message is sent to all consoles to which route codes 1 or 11 are assigned. For more detailed information regarding route codes, refer to the IBM publication Routing and Descriptor Codes, GC38-1102. Optionally, the message can also be sent to the CONTROL-M/Enterprise Manager user. This is described in "Shouting to CONTROL-M/Enterprise Manager" on page 473.

Specify TO=OPER2[-n] to send a highlighted, unrollable message to the operator console (route code n). If the n value is omitted, the message is sent to all consoles to which route codes 1 or 11 are assigned. For more detailed information regarding route codes, refer to the IBM publication Routing and Descriptor Codes, GC38-1102. Optionally, the message can also be sent to the CONTROL-M/Enterprise Manager user, as described in the following section, "Shouting to CONTROL-M/Enterprise Manager".

Specify TO=TSO-id or T-id to send the message to a groupid or logonid. The Shout facility first searches the IOA Dynamic Destination table for the specified ID. If the table contains an entry (groupid) that matches the value, the content of the entry is used as the target for the shouted message. (The entire TO field is used. Therefore, when directing the message to a remote user, do not append Nn or Mm. Instead, do this in the IOA Dynamic Destination Table itself. For more information, see the discussion of Destination Tables in the *INCONTROL for z/OS Administrator Guide*.)

If no matching ID is found in the Dynamic Destination table, the Shout facility assumes the specified ID is a logonid. It then creates a TSO message that it hands over to MVS. MVS then sends the message to that logonid. (If the logonid does not exist, MVS cannot send the message, but no error message is generated.) When a second value is used, the message is sent to the TSO logonid in the specified computer or node (machine ID). To determine the machine ID under JES2, specify JES command SD MEMBER.

Specify TO=U-M: mail-name-prefix to send the message by e-mail to the recipient identified by the prefix. The full mail name address is supplied by the MAILDEST table in the IOA PARM library. For more information about mail destinations, see the *INCONTROL for z/OS Administrator Guide*. The MAILDEST table also includes DFLTSFFX, the mail address suffix, for example: @MAIL.DOMAIN.COM, the SMTP STC name and the HOSTNAME. If installation parameter ATTSYSOT=Y, the job's SYSOUT will be attached to the e-mail message.

Specify TO=U-S:snmp_dest to send the SNMP trap (message) to the recipient identified by snmp_dest. This variable (snmp_dest) can be any of the following:

- a host name
- an IP address
- a nickname defined in the SNMPDEST table
- a group name defined in the SNMPDEST table

For more information about mail destinations, see the *INCONTROL* for z/OS *Administrator Guide*.

#### **Shouting to CONTROL-M/Enterprise Manager**

For CONTROL-M to be able to shout to CONTROL-M/Enterprise Manager, the following conditions must be satisfied at the site:

- 1. CONTROL-M/Enterprise Manager must be installed and the ECS parameter must be set to Y in the IOAPARM member in the IOA PARM library.
- 2. File MG2 (the CONTROL-M/Enterprise Manager Shout File) must be defined.
- 3. The following parameters in the IOAPARM member in the IOA PARM library must be defined according to how messages must be targeted to CONTROL-M/Enterprise Manager:
  - If TO=OPER and TO=OPER2 messages must be sent to CONTROL-M/Enterprise Manager, set the OPER2ECS parameter to Y (Yes). Otherwise, set it to N (No).

#### When OPER2ECS is set to Y:

- If these messages must also be sent to the MVS operator console, set the OPER2CON parameter to Y (Yes).
- If these messages must not also be sent to the MVS operator console, set the OPER2CON parameter to N (No).
- If TO=U-ECS messages must be sent to CONTROL-M/Enterprise Manager, set the ECS2ECS parameter to Y (Yes); otherwise, set it to N (No). Regardless of the value of this parameter, these messages are (also) sent to CONTROL-M and the IOA Log.

Once the above conditions are satisfied, messages can be shouted to CONTROL-M/Enterprise Manager by specifying a destination of TO=OPER or TO=OPER2 (without a route code qualifier) or TO=U-ECS.

Such messages are then placed by CONTROL-M in the M2G file. Once the shouted message is in the M2G file, the CONTROL-M Application Server reads the file and sends the message to the CONTROL-M/Enterprise Manager user.

#### The URGN Subparameter

The URGN value indicates the urgency level of the message.

In addition, if the destination is USERID-userid (or U-userid), the user can control, according to urgency, which messages are displayed when the IOA Log file is accessed. Urgent and very urgent messages are highlighted on the screen. For more details, see "IOA Log Facility" on page 296

#### Differences between SHOUT and DO SHOUT

SHOUT and DO SHOUT statements have the following differences:

■ A DO SHOUT statement is applied only if the accompanying ON criteria are satisfied. Therefore a DO SHOUT statement does not contain subparameters for specifying when to perform the shout.

By contrast, a SHOUT statement requires that a value be specified, in the WHEN subparameter, indicating when to shout the message. Messages can be shouted when the job ends OK or NOTOK, when the job is late for submission or completion, or when the job runs too long.

- A SHOUT statement appears in each job scheduling definition. A DO SHOUT statement does not appear unless specified. To specify a DO SHOUT statement, type SHOUT in an empty DO field and press Enter.
- The SHOUT URGN subparameter is equivalent to the DO SHOUT URGENCY subparameter.

The SHOUT MS subparameter is equivalent to the DO SHOUT subparameter.

## **Examples**

#### Example 1

If the job finishes executing OK, write a message to the IOA Log file under the specified user ID:

```
MEMNAME GPLSP0007

DAYS 01,15

SHOUT WHEN OK TO U-SHIFTMNGR URGN R

MS I HAVE FINISHED FOR TONIGHT
```

The message is written to the log under CONTROL-M userid-SHIFTMNGR.

### **Example 2**

When IMS is not active, send a message to all operators.

Figure 296 SHOUT Parameter Example 2

JOB: IMSPROD LIB CTM.PROD.SCHEDULE	TABLE: IMSPROD
COMMAND ===>	SCROLL===> CRSR
	=======================================
OUT	
AUTO-ARCHIVE Y SYSDB Y MAXDAYS MAXRUNS	
RETENTION: # OF DAYS TO KEEP 030 # OF GENERATIONS TO KEEP	
SYSOUT OP (C,D,F,N,R)	FROM
	FROM
STEP RANGE FR (PGM.PROC) . TO	
ON PGMST PROCST CODES	A/0
DO SHOUT WHEN OK TIME + DAYS TO OPER2	URGN R
MS ***** IMS IS NOT ACTIVE *****	UKUN K
SHOUT WHEN NOTOK TIME + DAYS TO OPER2	URGN R
MS ****** IMS IS NOT ACTIVE - ENDED ABNORMALLY ******	
SHOUT WHEN TIME + DAYS TO	URGN
MS	
===== >>>>>>>>>>>> END OF SCHEDULING PARAMETERS <<<<<<	<<<<<<<< =====
COMMANDS: EDIT, DOC, PLAN, JOBSTAT	11.17.00

If IMS finishes executing, an unrollable message is sent to the operator. A different message is sent if IMS terminates abnormally.

#### **Example 3**

If the job is not run because of a JCL error, notify the user who sent the job.

Figure 297 SHOUT and DO SHOUT Example

OUT		.,				
	SYSDE					
RETENTION: # SYSOUT OP ((	OF DAYS TO KEEP	U3U #	UF GENER	CALLONS	TO KEEP	FROM
MAXRERUN F					INTERVAL	
	FR (PGM.PF	OC)			TO	
	EP PROCST					A/0
DO SHOUT	TO TS0-U0014		URGENCY	U		
	ERROR IN SALARY	JOB ***				
DO	DDOOGT	0.05				
DO PGMST DO	PROCST	COL	JE2			A/0
SHOUT WHEN MS	TIME	+	DAYS	T0		URGN
===== >>>>>>>>	>>>>> END C	F SCHEDU	LING PAF	AMETER	s <<<<<<	<<<<<< =====

An urgent message is sent to the user ID that requested the job.

#### **Example 4**

Perform a LATE shout after the New Day time has passed and a new working day has begun.

Assume the following:

- New Day time at the site is 0600.
- A job, LONGWAIT, is ordered at 1400.
- The job LONGWAIT must shout if it has not finished executing by 0700 on the following day.

However, the New Day process automatically cancels any shout requirements of jobs ordered on any previous day.

There are two ways to achieve the required LATE shout:

#### Method 1

- 1 Create a DUMMY job scheduling definition named TRIGGER, containing the following elements:
  - The TIME FROM parameter is set to 1400.
  - The OUT parameter is set to TRIGGER-SHOUT.
- 2 Create a DUMMY job scheduling definition named SHOUT. This job must be ordered at the New Day time, and must contain the following elements:
  - The TIME FROM parameter is set to 0700.
  - The TIME UNTIL parameter is set to 1300.
  - The MAXWAIT parameter is set to 02.
  - The IN parameter is set to TRIGGER-SHOUT.
  - The SHOUT parameter is set to WHEN LATESUB 0700.
- **3** Add to the LONGWAIT JOB an OUT condition, TRIGGER-SHOUT, that deletes the TRIGGER-SHOUT condition when it ends.

This procedure works as follows:

- The IN condition in the SHOUT job prevents it from executing between 0600 and 1359.
- At 1400 the TRIGGER job adds the IN condition.
- The TIME FROM and UNTIL parameter values prevent the SHOUT job from running until after the next New Day procedure, but the MAXWAIT parameter value ensures that the job remains on the Active Jobs file for the following day.
- At 0700 on the following day
  - if the LONGWAIT job has ended, it has removed the IN condition required before the SHOUT job can run, so that there is no false shout at 0700
  - if the LONGWAIT job has not ended, the SHOUT job runs, and the shout is produced as required

#### Method 2

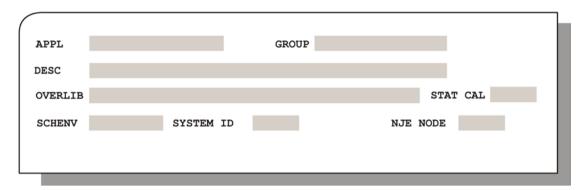
Use IOA Exit 34, and set values for the SET VAR parameter in the job scheduling definition.

For more information, see the IOAX034H sample exit in the IOA SAMPEXIT library.

# STAT CAL: General Job Parameter

Name of a periodic calendar that will be used to gather average runtime statistics for the job, based on a period.

Figure 298 STAT CAL Parameter Format



Optional. Valid values are any valid periodic calendar name consisting of from 1 through 8 characters.

For more information on defining, modifying, and using periodic calendars, see "IOA Calendar Facility" on page 306.

#### **General Information**

As part of the post-processing for each job, CONTROL-M for z/OS determines the elapsed run time of the job. All accumulated information regarding job execution, including the elapsed run time, is written to the IOA Log file. If a STAT CAL calendar is specified in the job scheduling definition, unlike other calendars (DCAL, WCAL, or CONFCAL), it must exist in the IOA Calendar library at the time that the job is ordered or forced.

Periodically, a statistics utility may be used to scan and analyze the IOA Log file. This utility gathers information about the start time of each job, its elapsed run time, CPU utilization time, and so on. The utility places this information in the Statistics file, where averages of these values can be maintained for each job.

For more information on the Statistics file, see "Statistics Screen" on page 238.

When a job is ordered, CONTROL-M takes the average run time of this job from Statistics file and places it in the job record in the Active Jobs file. CONTROL-M then uses this average run time to calculate the anticipated ELAPSE time, that is, the job execution time, of the job, and hence the Due In time of the job.

If the STAT CAL parameter is not used to specify a periodic calendar, the statistics relating to a job are based on all run times of the job.

If the STAT CAL parameter is present, statistics for the job are based on an average of all runtimes for the indicated period on the date on which the job is ordered or forced.

Further information is available in the Active Jobs file Zoom screen (Screen 3.z), which contains the STAT CAL PERIOD field. This is a read-only field that may contain one alphabetical character when the job has run. This character identifies the actual days within the CONTROL-M periodic calendar that were used in calculating statistics relating to the job.

By using the STAT CAL parameter together with the information displayed in the STAT CAL PERIOD field, you can obtain more precise statistical information about the running of the job, as shown in the following example.

#### **Example**

Assume that a job runs daily, weekly, and monthly, and that the STAT CAL parameter identifies a periodic calendar that contains a number of months each specified in a manner similar to the following:

In this example, the job runs daily in Period D, weekly in Period W, and monthly in Period M.

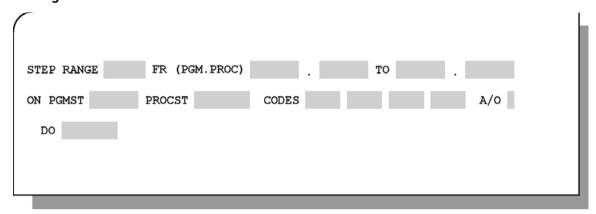
If the job runs on the 3rd of the month, its statistics are collected for Period W. If it runs on the 6th of the month, its statistics are collected for Period D, and so on.

When displaying the Statistics Panel (Screen 3.S), the statistics for daily, weekly, and monthly runs of the job are grouped under the D, W, and M PER(iod), as appropriate.

# **STEP RANGE: Post-Processing Parameter**

Step range in the job that can be used in an ON PGMST statement. For more information, see "ON Statements: Post-Processing Parameter" on page 537.

Figure 299 STEP RANGE Parameter Format



Optional. STEP RANGE consists of the subparameters described in Table 210.

**Table 210 STEP RANGE Subparameters** 

Subparameter	Description			
STEP RANGE	Name for the range. A name of 1 through 7 characters can be specified. Only trailing blanks are allowed in this field.			
FR (PGM,PROC)	First <i>pgmstep</i> or <i>pgmstep,procstep</i> in the range.			
	<b>Note:</b> <i>pgmstep</i> is the step name in the EXEC statement that identifies the program to be executed:			
	//pgmstep EXEC PGM=pgmname			
	<pre>procstep is the step name in the EXEC statement that invokes the procedure: //procstep EXEC procname</pre>			
	<i>pgmstep</i> values and <i>procstep</i> values can each be 1 through 8 characters in length.			
ТО	Last <i>pgmstep</i> or <i>pgmstep,procstep</i> in the range. For more information, see the note to the preceding subparameter, FR.			
	<b>Note:</b> The TO subparameter is optional. If blank, its value defaults to the last step in the job.			

#### **General Information**

Whenever a STEP RANGE statement is specified, it eliminates the need to define separate ON PGMST, ON PROCST, and ON CODES statements and accompanying DO actions for each step in the range. The defined STEP RANGE name can be used, without redefining the range, in subsequent ON PGMST, ON PROCST, and ON CODES statements, by specifying the step range name, preceded by an asterisk, in the ON PGMST field. For more information on ON PGMST and ON PROCST, see "ON Statements: Post–Processing Parameter" on page 537. For more information on ON CODES, see "CODES Values" on page 550.

Any number of step ranges can be specified. After entering a STEP RANGE parameter, another STEP RANGE parameter line is automatically displayed.

If the EXCLUDED STEP RANGE facility is activated (that is, EXSTPRNG=Y in the CTMPARM parameter) and the name of the step range starts with a minus sign, then that step range defines an EXCLUDED STEP RANGE (all steps in the job, excluding the steps from one step to another).

#### For example:

```
STEP RANGE -ERANG1 FR (PGM.PROC) STEP10 . TO STEP20
```

defines a step range called -ERANG1, which contains all the job steps except the steps from step STEP10 to step STEP20.

#### **Examples**

Triggers a SHOUT if any step, except step STEP3 through step STEP6, finishes with code 16.

```
STEP RANGE -ERANG1 FR (PGM.PROC) STEP3 . TO STEP6
ON PGMST *-ERANG1 PROCST CODES CO016
DO SHOUT ....
```

Triggers a SHOUT if every job step, except those from step STEP6 to step STEP9, finishes with a code greater than 8.

```
STEP RANGE -ERANG2 FR (PGM.PROC) STEP6 . TO STEP9
ON PGMST *-ERANG2 PROCST +EVERY CODES >C0008
DO SHOUT ...
```

Triggers a SHOUT if every job step, except step STEPA, finish with a zero code.

```
STEP RANGE -ERANG3 FR (PGM.PROC) STEPA . TO STEPA ON PGMST *-ERANG3 PROCST +EVERY CODES CO000 DO SHOUT ...
```

# §Restart§ Using All Runs of a Job Including Restarts

When processing ON blocks, CONTROL-M can incorporate the results of all previous runs and restarts, filtering them for jobs restarted with the parameters RESTART, RECAPTURE CONDITION and/or ABEND CODES. CONTROL-M/Restart searches previous runs to determine which steps must be considered part of the restarted job.

For example, if one step finished successfully during its original run and another step finished successfully after a restart, the ON block check for the successful finish for both steps produces a TRUE result and the ON statement is satisfied.

Activation of this facility requires that the ALLRUNS parameter in the CTRPARM parameter be set to YES. When activated, this facility may apply to any specified step, step range, or to step value +EVERY.

### **Example**

Define program steps STEP20 through STEP29A as step range DF2. If any of these steps produce any system or user abend (except user abend U2030), rerun the job and shout a message to TSO-P43.

Figure 300 STEP RANGE Parameter Example

```
JOB: PRDKPLO1 LIB CTM.PROD.SCHEDULE
                                                       TABLE: PRODKPL
  OUT
  AUTO-ARCHIVE Y
                      SYSDB Y MAXDAYS MAXRUNS
  RETENTION: # OF DAYS TO KEEP 030 # OF GENERATIONS TO KEEP
  SYSOUT OP (C,D,F,N,R)
MAXRERUN RERUNMEM
                                                              FROM
                                              INTERVAL
                                                           FROM
  STEP RANGE DF2 FR (PGM.PROC) STEP20 . TO STEP29A . STEP RANGE FR (PGM.PROC) . TO .
  STEP RANGE FR (PGM.PROC) . TO .
ON PGMST *DF2 PROCST CODES S**** U**** NU2030
                                                              A/0
   DO RERUN
   DO SHOUT TO TSO-P43
                                  URGENCY R
    = JOB PRDKPLO3 ABENDED, THE JOB IS RERUN
   D0
  ON PGMST
                               CODES
                                                              A/0
   D0
  SHOUT WHEN
                    TIME
                                   DAYS
                                                            URGN
  ===== >>>>>>>>>> END OF SCHEDULING PARAMETERS <<<<<<<<<<
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
```

# **SYSOUT: Post-Processing Parameter**

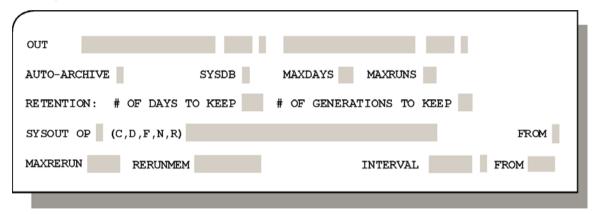
Controls handling of job output after the job ended OK.

#### **NOTE**



SYSOUT and DO SYSOUT statements are similar, but not identical. The differences are outlined below in "Differences Between SYSOUT and DO SYSOUT" on page 650.

Figure 301 SYSOUT Parameter Format



Optional. SYSOUT consists of the subparameters described in Table 211.

**Table 211 SYSOUT Subparameters** 

Subparameter	Description		
OP	<ul> <li>Sysout option code. Mandatory. Valid values are:</li> <li>F - Copy the job output to file.</li> <li>D - Delete (purge) the job output.</li> <li>R - Release the job output.</li> <li>C - Change the class of the job output.</li> <li>N - Change the destination of the job output.</li> </ul>		
sysout_data	Relevant sysout data. Mandatory and valid only if the specified OP value is F, C, or N. Valid values depend on the OP value, as follows:  F - File name C - New class (1 character). An asterisk (*) indicates the original MSGCLASS of the job N - New destination (1 through 8 characters)		
FROM	FROM class. Optional. Limits the sysout handling operation to only those sysouts from the specified class.  Note: If a FROM class is not specified, all sysout classes are treated as a single, whole unit.		

#### **General Information**

When a job ends OK, the CONTROL-M monitor, unless otherwise instructed, leaves the job sysout in HELD class in the output queue.

The SYSOUT parameter is used to request additional handling of these held sysouts when the job ends OK.

The CONTROL-M monitor sends all sysout handling requests to JES, which processes the instructions. If, however, the copying of sysouts to a file is requested (option F), CONTROL-M requests the sysouts from JES and then CONTROL-M directly writes the sysout to the file.

Only one SYSOUT statement can be defined in a job scheduling definition. To specify additional sysout handling instructions in addition to the one SYSOUT statement, define appropriate DO SYSOUT statements:

DO SYSOUT statements are activated when their accompanying ON step and code event criteria are satisfied. To define DO SYSOUT statements that act like a SYSOUT statement, that is, those that operate only when the job ends OK, define their accompanying ON statement with PGMST set to ANYSTEP and CODES set to OK.

For more information, see "ON Statements: Post–Processing Parameter" on page 537, and "DO SYSOUT: Post-Processing Parameter" on page 477.

The interrelationship between multiple sysout operations, as in SYSOUT and DO SYSOUT statements, is described in "Multiple Sysout Operations" on page 647.

### **Sysout Handling Operations**

Which sysouts are affected by sysout handling operations depends on whether the sysouts are under JES2 or JES3, as follows:

- Under JES2, operations are performed on all of the held sysouts, or held portions of sysouts, of the job, unless otherwise restricted to a specific FROM class by the FROM subparameter.
- Under JES3, operations are performed only on the sysouts of the job in the CONTROL-M held class, which is specified in the CONTROL-M installation parameter HLDCLAS.

Sysout handling operations are listed below:

■ Copying sysouts to a file (OPT=F)

The job sysouts are copied (not moved) to the file specified in the data subparameter.

The file name specified in the data subparameter can contain AutoEdit System variables, and/or user-defined AutoEdit variables, which are defined in the job scheduling definition or the IOA Global Variable database, or which are loaded into AutoEdit cache. If the AutoEdit variables cannot be resolved, the sysout is not copied.

CONTROL-M allocates the file with DISP=(NEW,CATLG,DELETE) using the unit and space attributes specified in the CONTROL-M installation parameters. While the block size (BLKSIZE) is automatically calculated by CONTROL-M, the logical record length (LRECL) is copied from the input SYSOUT file. The maximum LRECL allowed is 256 characters.

Sysouts can be archived by copying them to a file. However, to reduce overhead, this method is recommended only for small sysouts.

■ Deleting sysouts (OPT=D)

The job sysouts are deleted (purged) from the output queue.

#### _ NOTE -



This operation works on all sysouts under JES2 or JES3, regardless of held status or class, unless otherwise restricted by the FROM subparameter.

■ Releasing sysouts (OPT=R)

The job sysouts are released for printing.

■ Changing the class of sysouts (OPT=C)

The job sysouts are changed to the output class specified in the data subparameter. Ensure that you specify a meaningful target output class.

Note the following points:

— Changing a sysout class to a non-held class does not release the sysout because the sysout attributes do not change (due to JES logic).

— To ensure that the sysout is released, use DO SYSOUT statements to release the sysout after changing its class. For example,

```
DO SYSOUT OPT C PRM R FRM A DO SYSOUT OPT R PRM FRM A
```

- Changing a sysout class to a dummy class does not purge the sysout because the sysout attributes do not change (due to JES logic).
- To save the original MSGCLASS of the job and to restore it at output processing time, specify a data value of *. The sysouts are changed to the original class of the job.
- Moving sysouts to a new destination (OPT=N)

The job sysouts are moved to the output destination specified in the data subparameter. Ensure that you specify a meaningful target output destination.

#### **Multiple Sysout Operations**

If multiple SYSOUT or DO SYSOUT operations are not specified for the same FROM class, the order in which the operations are performed is not significant.

However, if different SYSOUT or DO SYSOUT operations affect the same FROM class, or if multiple operations are specified without a FROM class, the order and method of implementation is significant.

CONTROL-M merges different operations for the same FROM class into a combined instruction to JES. Likewise, CONTROL-M merges different operations without a FROM class into a combined instruction to JES.

Operations without a specified FROM class treat the entire held sysout as a whole unit, and are therefore not merged with sysout handling requests for a specific FROM class.

JES does not necessarily process multiple sysout handling instructions in the order they are issued by CONTROL-M. Therefore, the processing results can vary if the merged instructions to JES include both FROM equals a specified class and FROM equals blank.

BMC Software therefore recommends that a job scheduling definition not contain both "FROM class" and "no FROM class" sysout handling instructions, which becomes operational under the same situations.

When CONTROL-M merges a set of operations into a combined instruction, some operations override or cancel other operations, and some operations are performed along with other operations. This is described below.

#### **Operation Merging and Performance**

CONTROL-M performs all copy to file operations (option F) first.

After performing all copy to file operations, CONTROL-M merges all operations performed on a specific FROM class.

After merging operations on specific FROM classes, CONTROL-M merges the operations performed on the sysout as a whole (that is, subparameter FROM is set to blank).

CONTROL-M then passes the merged sets of instructions to JES for processing.

Generally, DO SYSOUT operations override, or are performed along with, SYSOUT statements.

The following chart and the accompanying numbered explanations indicate the result of merging SYSOUT and DO SYSOUT statements. Note the following points about the chart:

- Operations are indicated by their symbols (F,D,R,C,N), at the top and side of the chart. The operations at the top of the chart represent DO SYSOUT operations. The operations at the side of the chart represent SYSOUT operations.
- Merging and processing operations are grouped, and explained, based on operation type.

Groups are delimited by lines, and are numbered (from 1 through 4). Within each group, operations are delimited by periods.

Explanations of each group are provided, by number, following the chart.

■ The handling of the combination of operations is generally reflected in the chart by a single operation code (such as D) or pair of operation codes (such as FR).

In some cases, the operations are merged. This is indicated by the word "merged."

Operations are explained in the numbered descriptions that follow the chart.

DO SYSOUT  $\mathbf{F}$ D C N R F FD  $\mathbf{F}\mathbf{F}$ FC FNFR SYSOUT  $\mathbf{C}$ D FD D R N merged R FR D merged merged DO SYSOUT merged  $\mathbf{C}$ FC D R  $\mathbf{C}$ DO SYSOUT merged merged N FND N

Figure 302 Merging SYSOUT and DO SYSOUT Statements

The order of precedence in which CONTROL-M processes or merges operations is as follows:

#### 1. SYSOUT=F and DO SYSOUT=F

Copy to file operations are performed first, directly by CONTROL-M, for both SYSOUT and DO SYSOUT statements, whether FROM class is specified or not. Then, other operations are performed.

#### 2. DO SYSOUT=D (Delete)

This operation supersedes all SYSOUT operations, except copy to file operations described above. Superseded operations are ignored (that is, not performed).

#### 3. DO SYSOUT=R, C, or N, accompanied by a SYSOUT D (Delete) request

The DO SYSOUT statement is performed, and the SYSOUT delete request is ignored.

4. SYSOUT or DO SYSOUT combinations of R, C and N

In general, combinations of R, C, and N requests are merged, that is, they are all performed. The exceptional cases are described below:

— For DO SYSOUT=R (Release job output) accompanied by a SYSOUT C (Change class) request:

Perform just the DO SYSOUT R request and ignore the SYSOUT C request.

— For C (Change class) requests from both a SYSOUT and a DO SYSOUT statement:

Perform just the DO SYSOUT request and ignore the SYSOUT request.

— For N (New Destination) requests from both a SYSOUT and a DO SYSOUT statement:

Perform just the DO SYSOUT request and ignore the SYSOUT request.

### Differences Between SYSOUT and DO SYSOUT

SYSOUT and DO SYSOUT statements have the following differences:

- The SYSOUT statement is applied only if the job ends OK. DO SYSOUT statements are associated with accompanying ON statements and are applied only if the accompanying ON step and code criteria are satisfied.
- A SYSOUT statement is displayed in each job scheduling definition. A DO SYSOUT statement is not displayed unless requested. To request a DO SYSOUT statement, type SYSOUT in an empty DO field and press **Enter**.
- Only one SYSOUT statement can be defined in the job scheduling definition. An unlimited number of DO SYSOUT statements can be requested.
- The SYSOUT OP subparameter is equivalent to the DO SYSOUT OPT subparameter.
- The SYSOUT data subparameter is equivalent to the DO SYSOUT PRM subparameter.
- The SYSOUT FROM subparameter is equivalent to the DO SYSOUT FRM subparameter.

### **Examples**

### **Example 1**

Delete the sysout after the job has finished executing OK:

```
MEMNAME EBMANT1
DAYS 1,15
SYSOUT OP D (C,D,F,N,R)
```

#### **Example 2**

If the job finishes OK, reroute the sysout to printing class A.

#### Figure 303 SYSOUT Parameter – Example 2

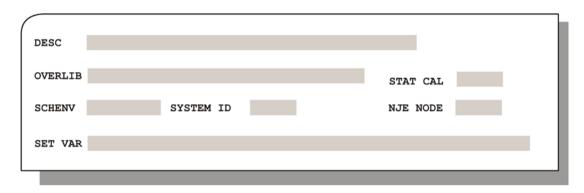
```
JOB: EBDMANT2 LIB CTM.PROD.SCHEDULE
                                                             TABLE: EBDPROD
COMMAND ===>
                                                             SCROLL===> CRSR
  ΙN
  CONTROL
  RESOURCE
  PIPE
  FROM TIME + DAYS UNTIL TIME +
DUE OUT TIME + DAYS PRIORITY SAC
                                                    DAYS
                                PRIORITY SAC CONFIRM
  TIME ZONE:
  OUT
  AUTO-ARCHIVE Y SYSDB Y MAXDAYS MAXRUNS
  RETENTION: # OF DAYS TO KEEP 030 # OF GENERATIONS TO KEEP SYSOUT OP C (C,D,F,N,R) A
                                                                    FROM
 MAXRERUN RERUNMEM
STEP RANGE FR (PGM.PROC)
ON PGMST PROCST CODES
                                                   INTERVAL
                                                                 FROM
                                                     T0
                                                                     A/0
   D0
  SHOUT WHEN TIME + DAYS TO
                                                                  URGN
  ---- >>>>>>>>> END OF SCHEDULING PARAMETERS <<<<<<<<<<<
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
```

# **SYSTEM ID: General Job Parameter**

In JES2, the identity of the system in which the job must be initiated and executed.

In JES3, the identity of the processor on which the job must execute.

Figure 304 SYSTEM ID Parameter Format



SYSTEM ID specifies a system identity of 1 through 4 characters. Only trailing blanks are allowed.

By default, the SYSTEM ID parameter is optional.

### **General Information**

The SYSTEM ID parameter has different effects, depending on which release of JES is in use.

If a value is specified for the SYSTEM ID parameter, it will not override any system identity specified in the job statement unless the OVERJCLM parameter in the CTMPARM library is set to Y.

If the task type is a started task, SYSTEM ID is protected. If the task type is changed from a job to a started task, SYSTEM ID is erased and protected.

### **Under JES2**

If CONTROL-M is running under JES2, the SYSTEM ID parameter is used to specify the JES2 system on which the job is to be initiated and executed.

If a value is specified for the SYSTEM ID parameter, the following JCL statement is generated:

/*JOBPARM SYSAFF=sys id

### **Under JES3**

If CONTROL-M is running under JES3, the SYSTEM ID parameter is used to specify the JES3 processor which is to execute the job.

If a value is specified for the SYSTEM ID parameter, the following JCL statement is generated:

//*MAIN SYSTEM=processor_id

### **Examples**

#### **Example 1: JES2**

The following is entered:

DESC		
OVERLIB		STAT CAL
SCHENV	SYSTEM ID SYS3	NJE NODE

The following statement is added to the JCL of the job:

/*JOBPARM SYSAFF=SYS3

and the job is executed on the JES2 system SYS3.

#### Example 2: JES3

The following is entered:

DESC			
OVERLIB			STAT CAL
SCHENV	SYSTEM ID	PRC3	NJE NODE

The following statement is added to the JCL of the job:

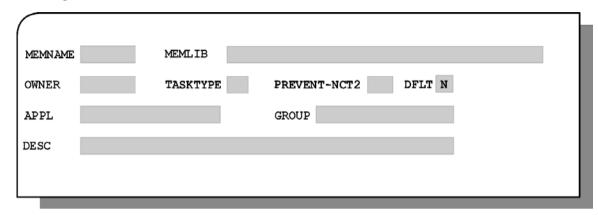
//*MAIN SYSTEM=PRC3

and the job is executed on processor PRC3.

# **TASKTYPE: General Job Parameter**

Type of task.

Figure 305 TASKTYPE Parameter Format



Mandatory. Valid TASKTYPE values are described in Table 212.

**Table 212 TASKTYPE Parameter Values** 

Value	Description
JOB	Batch job. Default
CYC	Cyclic job
STC	Started task (STC)
CST	Cyclic STC
EMR	Emergency job
ECJ	Emergency cyclic job
EST	Emergency STC
ECS	Emergency cyclic STC
WRN	Warning message

### **General Information**

The job scheduling definition can belong to one of three basic types of tasks:

- Job
- Started Task
- Warning Message

Jobs and started tasks can be "normal", that is, submitted or activated once, or cyclic. Furthermore, it is possible to define emergency versions of jobs and started tasks (normal and/or cyclic).

The three basic types of tasks are indicated by the TASKTYPE values described in Table 213:

Table 213 TASKTYPE Basic Type Values

Task	Values
Job:	JOB, CYC, EMR, ECJ
Started Task:	STC, CST, EST, ECS
Warning:	WRN

#### **Jobs and Started Tasks**

A regular job is submitted to the JES input queue for execution; it then waits in the queue like any job submitted under the operating system.

After the job finishes executing, CONTROL-M analyzes the results and determines what actions to take. The job is not submitted again unless the RERUN statement is performed. For additional information, see "MAXRERUN: Post–Processing Parameter" on page 516.

Started tasks differ from jobs in that they are not submitted to the queue; instead, they are invoked by an operator START command issued by CONTROL-M. For details on passing parameters to started tasks, see "MEMLIB: General Job Parameter" on page 523.

#### – NOTE



PREVENT-NCT2=Y cannot be specified for started tasks (STCs).

A cyclic job or a cyclic started task is recycled for additional possible executions after CONTROL-M has analyzed its execution results. The job or started task executes again only after the number of minutes specified in the INTERVAL parameter has passed since the last execution and the rest of its runtime scheduling criteria have been satisfied.

#### NOTE



BMC Software recommends that a cyclic job delete the prerequisite conditions that triggered its operation. Otherwise the job might continually be resubmitted.

If a cyclic job is executing at the time the New Day procedure is run, and the value of the job's MAXWAIT parameter has expired, the New Day procedure changes the job to a non-cyclic job and handles the job accordingly.

Use of the cyclic option precludes the use of RERUNMEM and DO RERUN parameters.

#### **Emergency Jobs and Started Tasks**



#### NOTE -

Emergency jobs and started tasks are supported for backward compatibility, but BMC Software recommends redefining them as regular jobs and started tasks that are activated by DO FORCEJOB statements. CONTROL-M/Restart users can also use a DO IFRERUN statement. The DO FORCEJOB statement is described in "DO FORCEJOB: Post–Processing Parameter" on page 444, and the DO IFRERUN statement in "§Restart§DO IFRERUN: Post–Processing Parameter" on page 447.

An emergency job or emergency started task can be used to overcome any irregularities in normal execution. The job remains in the Active Jobs file, waiting to be scheduled, until all regular jobs of the same GROUP finish executing OK and are checked by CONTROL-M. Then, when the emergency job is no longer needed, the job is automatically removed from the Active Jobs file. For additional information, see "MAXWAIT: Basic Scheduling Parameter" on page 519.

#### NOTE -



BMC Software recommends that the GROUP parameter be specified if you define emergency jobs. If it is not specified, the job may stay indefinitely in the Active Jobs file.

Emergency jobs can be filtered out of the job display in the Active Environment screen and filtered out of reports.

#### **Warning Messages**



#### - NOTE

The IOANOTE utility, which is described in the *INCONTROL for z/OS Utilities Guide*, can also be used to issue warning messages. BMC Software recommends that the IOANOTE utility be used in place of this tasktype wherever possible.

For tasktype WRN, warning messages are sent to the IOA Log file when the job is ordered. The messages are taken from the member specified in the MEMNAME parameter.

#### NOTE



A job defined with tasktype WRN is not placed in the Active Jobs file.

# **Examples**

### **Example 1**

Submit a regular job:

MEMNAME GNRLDR01 TASKTYPE JOB

### **Example 2**

Start a started task:

MEMNAME CICSPROD TASKTYPE STC

### **Example 3**

Start an emergency job:

MEMNAME RESTORE2
TASKTYPE EMR

### **Example 4**

Job OPERCOMP is a regular job.

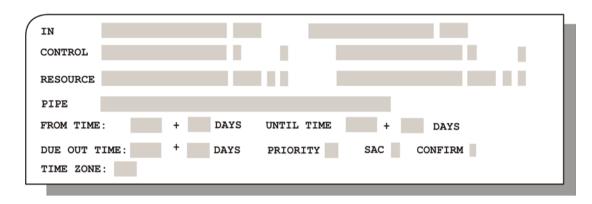
### Figure 306 TASKTYPE Parameter – Example 4

JOB: OPERCOMP LIB CTM.PROD.SCHEDULE COMMAND ===>	TABLE: OPER SCROLL===> CRSR
MEMNAME OPERCOMP MEMLIB CTM.PROD.JCL OWNER SYS1 TASKTYPE JOB PREVENT-NCT2 DFLT APPL OPER GROUP MAINTENANCE DESC JOB RUN ON THE 1ST OF THE MONTH OVERLIB	·
SCHENV SYSTEM ID SET VAR CTB STEP AT NAME TYPE DOCMEM OPERCOMP DOCLIB CTM.PROD.DOC	NJE NODE
DAYS 01	DCAL AND/OR
WDAYS MONTHS 1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- DATES	WCAL - Y 11- Y 12- Y
CONFCAL SHIFT RETRO Y MAXWAIT 00 D-CAT MINIMUM PDS DEFINITION ACTIVE FROM UNTIL	
IN COMMANDS: EDIT, DOC, PLAN, JOBSTAT	11.17.00

# TIME + DAYS: Runtime Scheduling Parameter

Time and day limits (FROM, UNTIL) for submitting a job or making a group active.

Figure 307 TIME + DAYS Parameter Format



Optional. This parameter consists of four subparameters:

- FROM TIME
- FROM DAYS
- UNTIL TIME
- UNTIL DAYS

Any or all subparameters may be specified.

The FROM TIME and UNTIL TIME subparameters can contain valid times in the format hhmm (where hh is hour and mm is minute). Character > can be entered in the UNTIL TIME field.

The FROM DAYS and UNTIL DAYS subparameters can contain valid numbers from zero through 120.

### **General Information**

The FROM TIME and FROM DAYS, and UNTIL TIME and UNTIL DAYS, subparameters define a window of opportunity for job submission. A job can only be submitted during the specified submission window.

### FROM DAYS and UNTIL DAYS are not entered

If either a FROM TIME or an UNTIL TIME subparameter is missing, that is, not specified, the New Day Processing time is used as the default for the missing value.

To create a submission window from a particular time until the New Day processing time, enter the desired FROM TIME and leave the UNTIL TIME subparameter blank.

#### **Example**



creates a submission window from 10:00 PM until the New Day Processing time.

To create a submission window from the New Day processing time until a particular time, enter the desired UNTIL TIME and leave the FROM TIME subparameter blank.

#### **Example**



creates a submission window from New Day processing time until 1:00 P.M. When both a FROM TIME and an UNTIL TIME value are specified, the relationship of New Day Processing time to these values (on a physical clock) determines the submission window. The logic is as follows:

If, when you move forward on the physical clock from the New Day Processing time, you arrive at the FROM TIME before you arrive at the UNTIL TIME, it means that New Day processing does not fall between the FROM TIME and the UNTIL TIME. In this case, the submission window runs from the FROM TIME to the UNTIL TIME, regardless of when the job was ordered.

#### **Example**

Assume a New Day Processing time of 8:00 A.M.:



creates a submission window from 2:00 P.M. until 10:00 P.M., a period of 8 hours.

#### **Example**

Assume a New Day Processing time of 10:00 P.M.:



creates a submission window from 11:00 P.M. until 5:00 A.M., a period of 6 hours.

#### **Example**

Assume a New Day Processing time of 10:30 P.M.:

FROM TIME 2300 + DAYS UNTIL TIME 2200 + DAYS

creates a submission window from 11:00 P.M. until 10:00 P.M., a period of 23 hours.

If, when you move forward on the physical clock from the New Day Processing time, you arrive at the UNTIL TIME before you arrive at the FROM TIME, it means that New Day processing falls between the FROM TIME and UNTIL TIME.

Batch jobs are frequently scheduled for submission from night until the morning. Therefore, when the New Day Processing time intervenes between the FROM TIME and the UNTIL TIME, it is likely that, following New Day Processing, the site still wants the job to be submitted up until the UNTIL TIME, without first waiting for the FROM time of the New Day.

For this reason, if New Day Processing comes between the FROM TIME and the UNTIL TIME, then regardless of when the job was ordered, the job is eligible for submission from both

- the FROM TIME until New Day Processing time
- New Day Processing time until the UNTIL TIME

The actual effect is that the submission window consists of all times except the interval from the UNTIL TIME until the FROM TIME.

#### **Example**

Assume a New Day Processing time of 4:00 A.M.:

FROM TIME 2300 + DAYS UNTIL TIME 0600 + DAYS

creates a submission window from 11:00 P.M. until 4:00 A.M. and from 4:00 A.M. until 6:00 A.M., giving a net submission window from 11:00 P.M. until 6:00 A.M. The job cannot be submitted from 6:00 A.M. until 11:00 P.M.

The character > in the UNTIL TIME subparameter indicates that CONTROL-M must attempt to submit the job at the FROM TIME if specified, and if this is not possible, CONTROL-M must submit the job as soon afterwards as possible, even at a later date (unless the MAXWAIT period has expired).

#### **Example**

Assume a New Day Processing time of 8:00 A.M.:

```
FROM TIME 2200 + DAYS UNTIL TIME > + DAYS
```

creates a submission window that begins at 10:00 P.M. If the job has not been submitted by the end of day, it can be submitted at any time from the beginning of the next day.

The FROM TIME subparameter is ignored when a job is rerun or restarted on a subsequent day.

Specifying the same time in both the FROM TIME and the UNTIL TIME subparameters has the same impact as entering no value in both fields.

#### **Example**

Submit the job after midnight:

#### Figure 308 FROM TIME Parameter Example

```
JOB: OPGENBKP LIB CTM.PROD.SCHEDULE TABLE: BACKUP
COMMAND ===> SCROLL===> CRSR
+-----
SCHENV SYSTEM ID NJE NODE
SET VAR
CTB STEP AT NAME TYPE
DOCMEM OPGENBKP DOCLIB CTM.PROD.DOC
DAYS DCAL
AND/OR
WDAYS WCAL
MONTHS 1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- Y 11- Y 12- Y
CONFCAL SHIFT RETRO Y MAXWAIT OO D-CAT
MINIMUM PDS
DEFINITION ACTIVE FROM UNTIL
CONTROL
RESOURCE
PIPF
FROM TIME 0000
               + DAYS
+ DAYS
                             UNTIL TIME
                                               DAYS
                             PRIORITY SAC CONFIRM
DUE OUT TIME
```

In this example, if the start time of the new workday is 6:00 A.M., this job can only be submitted between the hours of midnight and 6:00 A.M.

### FROM DAYS and UNTIL DAYS are entered

The FROM DAYS and UNTIL DAYS subparameters, together with the FROM TIME and TO TIME subparameters, can be used to define a window of opportunity for job submission that will take place on a future date, at a time more than 24 hours in the future.

If the FROM DAYS and UNTIL DAYS subparameters are entered, the job's submission window is calculated as follows:

- start date is the job's ODATE plus FROM DAYS
- end date is the job's ODATE plus UNTIL DAYS

If either the FROM TIME or UNTIL TIME subparameter is missing, that is, not specified, the New Day Processing time is used as the default for the missing value.

To create a submission window from a particular date and time until the date and time of New Day processing, enter the desired FROM TIME and leave UNTIL TIME blank.

For all examples in this section, assume an ODATE of September 5.

#### **Example**



creates a submission window from CONTROL-M working date September 6 at 10:00 P.M. until New Day Processing time on CONTROL-M working date September 8th.

To create a submission window from a particular date and time of New Day processing until a particular date and time, enter the desired UNTIL TIME and leave FROM TIME blank.

#### **Example**

FROM TIME (blank) + 1 DAYS UNTIL TIME 1300 + 3 DAYS

creates a submission window from New Day processing on CONTROL-M working date September 6 until 1:00 P.M. on CONTROL-M working date September 8.

When both a FROM TIME and an UNTIL TIME value are specified, the relationship of the New Day Processing time to these values (on a physical clock) determines the submission window. The logic is as follows:

If, when you move forward on the physical clock from the New Day Processing time, you arrive at the FROM TIME before you arrive at the UNTIL TIME, it means that New Day processing does not fall between the FROM TIME and UNTIL TIME. In this case, the submission window runs from the FROM TIME to the UNTIL TIME, regardless of when the job was ordered.

#### **Example**

Assume a New Day Processing time of 8:00 A.M.:

FROM TIME 1400 + 1 DAYS UNTIL TIME 2200 + 3 DAYS

creates a submission window from 2:00 P.M. on CONTROL-M working date September 6 until 10:00 P.M. on CONTROL-M working date September 8, a period of 56 hours.

#### **Example**

Assume a New Day Processing time of 10 P.M.:

FROM TIME 2300 + 1 DAYS UNTIL TIME 0500 + 3 DAYS

creates a submission window from 11:00 P.M. on CONTROL-M working date September 6 until 5:00 A.M. on CONTROL-M working date September 8, a period of 54 hours.

#### **Example**

Assume a New Day Processing time of 10:30 P.M.:

FROM TIME 2300 + 1 DAYS UNTIL TIME 2200 + 3 DAYS

creates a submission window from 11:00 P.M. on CONTROL-M working date September 6 until 10:00 P.M. on CONTROL-M working date September 8, a period of 47 hours.

If, when you move forward on the physical clock from the New Day Processing time, you arrive at the UNTIL TIME before you arrive at the FROM TIME, it means that New Day processing falls between the FROM TIME and UNTIL TIME.

Batch jobs are frequently scheduled for submission during the night. Therefore, when the New Day Processing time intervenes between the FROM TIME and the UNTIL TIME, it is likely that, following New Day Processing, the site still wants the job to be submitted up until the UNTIL TIME, without first waiting for the FROM TIME of the new day. For this reason, if the New Day Processing time comes between the FROM TIME and the UNTIL TIME, then regardless of when the job was ordered, the job is eligible for submission from both:

- the FROM TIME until the New Day Processing time
- the New Day Processing time until the UNTIL TIME

#### **Example**

Assume New Day Processing time of 4:00 A.M.:

FROM TIME 2300 + 1 DAYS UNTIL TIME 0600 + 3 DAYS

creates a submission window from 11:00 P.M. on CONTROL-M working date September 6 until 4:00 A.M. on CONTROL-M working date September 8, and from 4:00 A.M. on CONTROL-M working date September 8 until 6:00 A.M. on CONTROL-M working date September 8, giving a net submission window from 11:00 P.M. on CONTROL-M working date September 6 until 6:00 A.M. on CONTROL-M working date September 8.

The character > in the UNTIL TIME subparameter indicates that CONTROL-M must attempt to submit the job at the FROM TIME if specified, and if this is not possible, CONTROL-M must submit the job as soon afterwards as possible, even at a later date (unless the MAXWAIT period has expired).

On CONTROL-M screens 2 and 3, if the UNTIL TIME is >, the UNTIL DAYS subparameter can not be entered.

#### **Example**

Assume a New Day Processing time of 8:00 A.M.:

FROM TIME 2200 + 1 DAYS UNTIL TIME > + DAYS

creates a submission window that begins at 10:00 P.M. on CONTROL-M working date September 6. If the job has not been submitted by the end of the day, it can be submitted at any time from the beginning of the following day.

The FROM parameter is ignored when a job is rerun or restarted on a subsequent day.

Specifying the same time and day in both the FROM TIME/DAYS subparameters and the UNTIL TIME/DAYS subparameters has the same impact as entering no value in both fields.

#### **Example**

Submit the job after midnight:

#### Figure 309 TIME + DAYS Parameter Example

```
JOB: OPGENBKP LIB CTM.PROD.SCHEDULE TABLE: BACKUP
COMMAND ===> SCROLL===> CRSR
SCHENV SYSTEM ID NJE NODE
SET VAR
CTB STEP AT NAME TYPE
DOCMEM OPGENBKP DOCLIB CTM.PROD.DOC
DAYS DCAL
AND/OR
WDAYS WCAL
MONTHS 1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- Y 11- Y 12- Y
CONFCAL SHIFT RETRO Y MAXWAIT OO D-CAT
MINIMUM PDS
DEFINITION ACTIVE FROM UNTIL
ΤN
CONTROL
RESOURCE
PIPF
                 + 1 DAYS
FROM TIME 0000
                                UNTIL TIME
                                                     DAYS
              + DAYS
                                PRIORITY SAC CONFIRM
DUE OUT TIME
```

In this example, if New Day Processing time is 6:00 A.M., this job can only be submitted between the hours of midnight and 6:00 A.M. on CONTROL-M working date September 6.

### **Time Zone Considerations**

The TIME ZONE parameter affects the FROM TIME+DAYS and UNTIL TIME+ DAYS parameters. For details, please refer to "TIME ZONE: Runtime Scheduling Parameter" on page 667.

### **MAXWAIT Considerations**

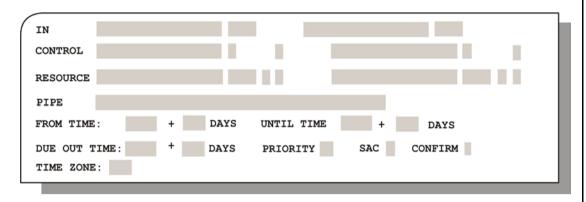
The FROM DAYS and UNTIL DAYS parameters affect the MAXWAIT parameter. For details, please refer to "MAXWAIT: Basic Scheduling Parameter" on page 519.

# **TIME ZONE: Runtime Scheduling Parameter**

Adjusts the values specified in the TIME parameter to those in a different time zone.

A related parameter is "TIME + DAYS: Runtime Scheduling Parameter" on page 659.

Figure 310 TIME ZONE Parameter Format



Optional. TIME ZONE specifies a time zone using three characters.

### **General Information**

The TIME ZONE parameter appears in the Job Scheduling Definition screen (Screen 2) and the Active Environment Zoom screen (Screen 3.z).

The TIME ZONE parameter uses three characters to define a time zone by reference to Greenwich Mean Time (UTC). This enables automatic adjustment of the times specified in some fields of the Job Scheduling Definition screen to the corresponding times in a time zone other than that in which the system is operating. The fields that are automatically adjusted are the following:

- FROM TIME+DAYS
- UNTIL TIME+DAYS
- DUE OUT TIME+DAYS
- SHOUT WHEN TIME+DAYS

If you set the TIME ZONE parameter appropriately, CONTROL-M calculates the corresponding times automatically, and the job runs only during the hours you require.

The three-character values used in the TIME ZONE parameter are defined in the TIMEZONE member in the IOA PARM library. A sample TIMEZONE member is provided, but you can edit the values as needed. For example, you can use "EST" or "NYC" instead of "G-5", which is the default value for US Eastern Standard Time.

#### - NOTE



Daylight Saving Time is defined differently in different time zones. For more information on Daylight Savings Time, see the CONTROL-M chapter in the *INCONTROL for z/OS Administrator Guide*.

### **Example**

You are running CONTROL-M in London, but want a job to run only when the New York Stock Exchange is open, between 0900 (9 A.M.) and 1600 (4 P.M.) in New York (US Eastern Standard Time). US Eastern Standard Time is five hours behind London time (GMT-5 hours).

- 1. In the Job Definition screen (Screen 2) or the Active Environment Zoom screen (Screen 3.Z) set the TIME FROM parameter to 0900 and the UNTIL parameter to 1600.
- 2. The TIMEZONE member defines GMT-5 hours as EST.
- 3. Set the TIME ZONE parameter in the same screen to EST. When you press the **Enter** key, the CONTROL-M interpretation of your specification is also displayed in the format (GMT*xhh*: *mm*)

#### where:

- $\blacksquare$  x is + (Plus) or (Minus)
- *hh* is the hours figure you specified
- mm is the minutes figure specified, either 00 or 30
- 4. The job will run between 2 P.M. and 9 P.M. at your site in London. These times correspond respectively to 9 A.M. and 4 P.M. in New York, the hours when the New York Stock Exchange is open. In other words, the job runs as if the TIME FROM was set to 1400 and UNTIL to 2100.

#### NOTE



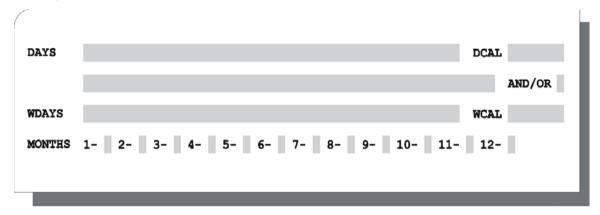
In the Job Scheduling Definition screen (Screen 2) and the Active Environment Zoom screen (Screen 3.Z), the times appear as you specified, as 0900 and 1600 respectively. In the Active Environment Why screen (Screen 3.?), the times appear with their converted values of 1400 and 2100 respectively.

# **WDAYS: Basic Scheduling Parameter**

Days of the week on which the job must be scheduled.

Related parameters are "DAYS: Basic Scheduling Parameter" on page 420, and "CONFCAL: Basic Scheduling Parameter" on page 402.

Figure 311 WDAYS Parameter Format



Optional. The WDAYS parameter specifies days of the week on which jobs must be scheduled, provided other basic scheduling criteria are met.

Values for WDAYS can be specified alone, or they can be specified together with a calendar specified in the WCAL subparameter. WDAYS and WCAL can also be specified together with DAYS and DCAL, which are described under "DAYS: Basic Scheduling Parameter" on page 420.

WDAYS subparameters are described in Table 214.

Table 214 WDAYS Subparameters (part 1 of 2)

Subparameter	Description
WDAYS	Days of each week in the month on which to schedule a job. (The months in which to order jobs are specified in the MONTHS parameter.) Various formats (described later) can be used to specify WDAYS; for example, 2 means the second day of the week, L2 means the day before the last day of the week.
	<b>Note</b> : At time of installation, the INCONTROL administrator selects either Sunday or Monday as the "first" day of the week. Your INCONTROL administrator can tell you whether the week begins on Sunday or Monday at your site.
	The first six days of the week are coded 1 through 6. The last day of the week is coded 0 (zero). All examples in this chapter assume Monday is the first day of the week. In these examples, Monday = 1, Tuesday = 2,, Saturday = 6, and Sunday = 0.

Table 214 WDAYS Subparameters (part 2 of 2)

Subparameter	Description
WCAL	Name of a calendar containing a predefined set of dates, referred to as working days, on which a job is to be scheduled. A specified value must be either a valid member name of 1 through 8 characters, or an * to indicate that the calendar specified in the CONFCAL parameter must be used for scheduling. For more information about how to define, use and modify calendars, see "IOA Calendar Facility" on page 306.
	<b>Note</b> : A calendar specified in WCAL does not have to exist when defining the WDAYS parameter. However, it must exist when the job is to be ordered.

Assuming all other basic scheduling criteria are met:

- When WDAYS are specified without WCAL, the job is scheduled on the specified days of the week.
- When WCAL is specified without WDAYS, the job is scheduled on the working days marked in the WCAL calendar.
- When WDAYS and WCAL are both specified, scheduling depends on the combination of working days defined in the calendar and the values or format of the WDAYS parameter (described below).
- When both DAYS and WDAYS criteria are specified, scheduling depends on the connecting AND/OR value specified. For more information, see subparameter AND/OR in Table 167 on page 420.

### **Valid Formats for WDAYS**

Valid formats for the WDAYS parameter, and how they relate to WCAL, are described below.

In the non-periodic scheduling formats described in Table 215, n is an integer from 1 through 6 or 0 (zero), where 1 = the first day of the week (Sunday or Monday, depending on the standard at your site) and 0 = the last day of the week (Saturday or Sunday).

- Multiple values can be expressed, separated by commas, in any order.
- WCAL must not contain the name of a periodic calendar.

**Table 215 Non-Periodic Scheduling Formats** 

Format	Description
ALL	All days of the week. If ALL is specified, other WDAYS values cannot be specified with it.  If a WCAL calendar is not defined, schedule the job on all days in the week.  If a WCAL calendar is defined, schedule the job only on the working days indicated in the calendar.
n,	Specific days of the week.  If a WCAL calendar is not defined, schedule the job on the specified days.  If a WCAL calendar is defined, schedule the job only when a day is defined as a working day both in the WDAYS parameter and in the WCAL calendar.
+n,	Days of the week in addition to the working days specified in the WCAL calendar. WCAL is mandatory.
-n,	Order the job on all days except the <i>n</i> th day from the beginning of the week. WCAL is mandatory.
>n,	Order the job on the indicated day if it is a working day in the WCAL calendar; otherwise, order the job on the next working day (within the next seven days ^a ) that is not negated by $a - n$ value in the parameter. This format is frequently used for holiday handling. WCAL is mandatory.
<n,< td=""><td>Order the job on the indicated day if it is a working day in the WCAL calendar; otherwise, order the job on the last previous working day (within the preceding seven days^a) that is not negated by a $-n$ value in the parameter. This format is frequently used for holiday handling. WCAL is mandatory.</td></n,<>	Order the job on the indicated day if it is a working day in the WCAL calendar; otherwise, order the job on the last previous working day (within the preceding seven days ^a ) that is not negated by a $-n$ value in the parameter. This format is frequently used for holiday handling. WCAL is mandatory.
Dn,	Order the job on the <i>n</i> th working day from the beginning of the week. WCAL is mandatory.
-D <i>n</i> ,	Order the job on all working days except the <i>n</i> th working day from the beginning of the week. WCAL is mandatory.
Ln,	Order the job on the <i>n</i> th working day counting from the end of the week. WCAL is mandatory.
-Ln,	Order the job on all working days except the <i>n</i> th working day counting backward from the end of the week. WCAL is mandatory.
DnWm,	(Where <i>m</i> = 1 through 6). If WCAL is defined, order the job on the <i>n</i> th working day of the <i>m</i> th week of the month. If WCAL is not defined, order the job on the <i>m</i> th appearance of the <i>n</i> th day of the week during the month. A maximum of eleven D <i>n</i> W <i>m</i> values can be specified. WCAL is optional.

^a If none of those seven days is a working day, the job is not ordered.

In the periodic scheduling formats described in Table 216

• n is any integer from 0 through 6, and i is any valid period identifier.

- WDAYS periodic identifiers are counted on a week by week basis. Calculations do not cross week boundaries, unlike DAYS periodic identifiers, which can cross month boundaries.
- The name of a periodic calendar must be specified in WCAL. For details concerning periodic calendars, see "IOA Calendar Facility" on page 306.
- A maximum of eight periodic values can be specified in any desired order.

**Table 216 Periodic Scheduling Formats** 

Format	Description
DnPi,	Order the job on the <i>n</i> th day of period <i>i</i> in each week, from the beginning of the week. An * can be entered as the <i>i</i> value to represent all periods.
-D <i>n</i> P <i>i</i> ,	Order the job on all days except the <i>n</i> th day of period <i>i</i> in each week, from the beginning of the week. An * can be entered as the <i>i</i> value to represent all periods.
LnPi,	Order the job on the <i>n</i> th day of period <i>i</i> in each week, counting backward from the last periodic day of the week. An * can be entered as the <i>i</i> value to represent all periods.
-LnPi,	Order the job on all days in period $i$ except the $n$ th day of period $i$ in each week, counting from the last periodic day of the week. An * can be entered as the $i$ value to represent all periods.

#### — WARNING



Before you use the  $P^*$  format of the periodic scheduling criteria, review example 11 below to ensure that you are aware of its proper functioning.

### **General Information**

Negative values take precedence over positive values when determining whether a job is scheduled on a certain date. If a negative value (that is, format -n, -Dn, -Ln, -DnPi, or -LnPi) in either the DAYS or WDAYS field prevents a job from being scheduled on a date, the job is not scheduled on that date even if a positive value (such as Ln) in a basic scheduling parameter would otherwise result in the job being scheduled on that date.

If periodic and non-periodic values are mixed when specifying the WDAYS parameter, processing depends on the calendar type specified in the WCAL parameter:

■ If a non-periodic calendar is specified in the WCAL parameter, only non-periodic values in the WDAYS parameter are processed; periodic values are ignored. In this case, negative periodic values (that is, -DnPi, -LnPi) are also ignored and do not supersede other values.

■ If a periodic calendar is specified in the WCAL parameter, all periodic values and the negative non-periodic value -n in the WDAYS parameter are processed; all nonnegative non-periodic values are ignored.

The WDAYS parameter cannot be used with the PDS and MINIMUM parameters.

### **Examples**

The examples in this chapter are based on the following assumptions:

- The current month is December, 2001.
- Working days are defined in calendar WORKDAYS, which contains the following working days (indicated by Y) for December, 2001.

 Periodic calendar PERIDAYS contains the following periodic definition for December 2001. These examples assume that all other days of this calendar are blank.

■ Start of the week is defined as Monday. Weeks start on the following dates in December 2001: 3rd, 10th, 17th, 24th, and 31st.

At the end of each example, asterisks in a December 2001 calendar indicate the days on which the job is scheduled.

#### **Example 1**

Schedule the job on every Sunday and Monday.

```
WDAYS 0,1
```

The job is scheduled on the days of the month indicated by an asterisk:

Figure 312 WDAYS Parameter Example 1

01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
SA	SU	МО	TU	WE	ТН	FR	SA	su	МО	TU	WE	ТН	FR	SA	SU	МО	TU	WE	ΤH	FR	SA	SU	МО	TU	WE	TH	FR	SA	SU	МО
	*	*						*	*						*	*						*	*						*	*

### **Example 2**

Schedule the job on all working days and on all Saturdays.

WDAYS	+6		
WCAL	WORKDAYS		

The job is scheduled on the days of the month indicated by an asterisk:

Figure 313 WDAYS Parameter Example 2

01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
SA	SU	МО	IJ	WE	ĭ	FR	SA	su	МО	IJ	WE	Ŧ	FR	SA	SU	МО	T	WE	픋	FR	SA	SU	МО	TU	WE	ΤH	FR	SA	SU	МО
*		*	*	*	*	*	*		*	*	*	*	*	*		*	*	*	*	*	*		*		*	*	*	*		*

### **Example 3**

Schedule the job on Sunday, if it is a working day. If Sunday is not a working day, schedule the job on the first preceding working day that is not a Friday.

```
WDAYS -5,<0
WCAL WORKDAYS
```

The job is scheduled on the days of the month indicated by an asterisk:

Figure 314 WDAYS Parameter Example 3

01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
SA	SU	МО	TU	WE	TH	FR	SA	su	МО	TU	WE	ТН	FR	SA	SU	МО	TU	WE	TH	FR	SA	SU	МО	TU	WE	TH	FR	SA	su	МО
					*							*							*							*				

#### **Example 4**

Schedule the job on Monday of the 1st week.

WDAYS	D1W1			

The job is scheduled on the days of the month indicated by an asterisk:

Figure 315 WDAYS Parameter Example 4

01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
SA	SU	МО	TU	WE	ТН	FR	SA	su	МО	TU	WE	ТН	FR	SA	SU	МО	TU	WE	ΤH	FR	SA	SU	МО	TU	WE	TH	FR	SA	SU	МО
		*																												

#### **Example 5**

Schedule the job on all working days except Mondays and Fridays.

WDAYS	-D1 -I1
WDAIS	- DI, - LI
LICAL	HODKDAVC
WCAL	WORKDAYS

The job is scheduled on the days of the month indicated by an asterisk:

Figure 316 WDAYS Parameter Example 5

01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
SA	SU	МО	TU	WE	ΤH	FR	SA	su	МО	TU	WE	ТН	FR	SA	SU	МО	TU	WE	Ħ	FR	SA	SU	МО	TU	WE	TH	FR	SA	SU	МО
			*	*	*					*	*	*					*	*	*					7	*	*				

#### **Example 6**

Each week, schedule the job on the 1st day of period A in that week, and on all days of period B except the second day of period B in any week.

WDAYS	D1PA,-D2PB		
WCAL	PERIDAYS		

The job is scheduled on the days of the month indicated by an asterisk:

Figure 317 WDAYS Parameter Example 6

01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
SA	SU	МО	TU	WE	Ŧ	FR	SA	su	МО	TU	WE	Ħ	FR	SA	SU	МО	J	WE	TH	FR	SA	SU	МО	TU	WE	Ħ	FR	SA	su	МО
		*		*					*		*					*		*					*		*					*

### **Example 7**

Schedule the job on each Monday and on the 1st day of the month.

DAYS	1
AND/OR	OR .
WDAYS	1

The job is scheduled on the days of the month indicated by an asterisk:

Figure 318 WDAYS Parameter Example 7

01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
SA	SU	МО	IJ	WE	ТН	FR	SA	SU	МО	IJ	WE	Ŧ	FR	SA	SU	МО	T	WE	TH	FR	SA	SU	МО	TU	WE	ΤH	FR	SA	SU	МО
*		*							*							*							*							*

### **Example 8**

Schedule the job on the 3rd day of the month provided it is a Monday.

DAYS	3	
AND/OR	AND	
WDAYS	1	

The job is scheduled on the days of the month indicated by an asterisk:

Figure 319 WDAYS Parameter Example 8

01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
SA	SU	МО	TU	WE	ТН	FR	SA	su	МО	TU	WE	ТН	FR	SA	SU	МО	TU	WE	ΤH	FR	SA	SU	МО	TU	WE	TH	FR	SA	SU	МО
		*																												

### **Example 9**

Schedule the job on the last Monday of the month.

```
DAYS L1,L2,L3,L4,L5,L6,L7
AND/OR AND
WDAYS 1
```

The job is scheduled on the days of the month indicated by an asterisk:

Figure 320 WDAYS Parameter Example 9

01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
SA	SU	МО	TU	WE	ТН	FR	SA	su	МО	TU	WE	ТН	FR	SA	SU	МО	TU	WE	크	FR	SA	SU	МО	TU	WE	TH	FR	SA	su	МО
																														*

#### **Example 10**

Schedule the job on the 1st, 7th and 15th day of the month if they are both Saturdays and working days. If the day of the month (1st, 7th, 15th) is not a Saturday, do not schedule the job. If the day of the month is a Saturday, but it is not a working day, schedule the job on the next working day.

```
DAYS 1,7,15
AND/OR AND
WDAYS 6
CONFCAL WORKDAYS
SHIFT >
```

The job is scheduled on the days of the month indicated by an asterisk:

Figure 321 WDAYS Parameter Example 10

01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
SA	SU	МО	ΤU	WE	ТН	FR	SA	su	МО	TU	WE	ТН	FR	SA	SU	МО	TU	WE	크	FR	SA	SU	МО	TU	WE	TH	FR	SA	SU	МО
		*														*														

#### **Example 11**

Schedule a job, on a weekly basis, on the first day of period C, using periodic calendar PER, in the month of August 2006. The first day of the week is defined as Monday (SWEEK=MON):

The months of July and August are defined in the periodic calendar PER as follows:

The job will be scheduled on the following days in August: Sunday the 27th and Monday the 28th.

The job is not scheduled on August 1st, because August 1st is contained in the week July 31 through August 6, and therefore, the first day of period C in that week is July 31.

The job is scheduled on August 27, because it is the first (and only) day in period C of the week August 21 through August 27. The job is also obviously scheduled on August 28.

■ Schedule a job, on a weekly basis, according to the criteria D1P* (the first day of any period), using periodic calendar PER (defined above), in the month of August 2006.

WDAYS D1P* WCAL PER

The job will be scheduled on August 7, 14, 21, and 28 only. The job will not be scheduled on August 6, 13, 20, or 27, because even though those dates contain the first day of a period in the respective week, the D1P* criterion has already been satisfied by a previous day (in a different period) in that same week.





# **CONTROL-M Event Manager (CMEM)**

This chapter includes the following topics:

Overview
Types of Events Managed by CMEM
Types of Actions That CMEM Can Perform
CMEM Rule Ordering, Triggering and Deactivation
CMEM AutoEdit Variables
On Spool Jobs
Support for ON DSNEVENT and ON STEP
CMEM Support for FTP
CMEM Support for IBM FTP692
Rule Parameters – Summary
Event Selection Parameters
General Parameters
Action Parameters
Parameter Descriptions
DESCRIPTION: General Parameter
DO statement: Action Parameter
DO COND: Action Parameter         700
DO FORCEJOB: Action Parameter
DO RULE: Action Parameter
DO SHOUT: Action Parameter
DO STOPJOB: Action Parameter
GROUP: General Parameter717
MODE: General Parameter
ON statement: Event Parameter
ON DSNEVENT: Event Parameter
ON JOBARRIV: Event Parameter
ON JOBEND: Event Parameter
ON STEP: Event Parameter
OWNER: General Parameter
RUNTSEC: General Parameter
THRESHOLD: Runtime Scheduling Parameter

# **Overview**

The CONTROL-M Event Manager (CMEM) facility enables CONTROL-M to perform specified actions in response to external events. External events are events in the system that occur outside the direct control of CONTROL-M, such as submission of a job not under the control of the CONTROL-M monitor.

The CMEM facility is an optional facility based on a monitor and a subsystem.

The CMEM facility utilizes sets of user-defined rules that specify events to monitor and actions to perform if a specified event occurs. These rules are defined online through the CMEM Rule Definition facility.

Multiple rules can be defined in a table (member) in a standard partitioned data set (library). Related rules are usually defined in the same table. Multiple tables can be defined in a library, and multiple CMEM rule libraries can be defined.

## **Types of Events Managed by CMEM**

The CMEM facility handles the following events. These can be specified in ON statements in the rule:

Table 217 Events handled by CMEM

Event	Description
DSNEVENT	Data set disposition, such as cataloged, deleted or kept, during step termination or dynamic decollation, or the occurrence of a NOT CATLGD 2 event, when a data set name is created in a job step but not cataloged because its name already exists in the MVS catalog. Specified in an ON DSNEVENT statement in the rule.
JOBARRIV	Arrival of a job on the JES spool from any source.  Examples are:
	■ jobs submitted by a TSO user or by CICS ■ jobs received over an NJE network
	Specified in an ON JOBARRIV statement in the rule.
JOBEND	Completion of a job regardless of its source. Specified in an ON JOBEND statement in the rule.
STEP	Termination of a job step. Specified in an ON STEP statement in the rule.

## **Types of Actions That CMEM Can Perform**

Any combination of the following actions can be performed when the specified event occurs. They are specified in DO statements in the rule:

add or delete a prerequisite condition

Prerequisite conditions can be added to or deleted from the IOA Conditions file. This may trigger the submission of jobs in the Active Jobs file. Specified in a DO COND statement in the rule.

force a job or table

A CONTROL-M scheduling table or individual job can be forced (that is, ordered to the Active Jobs file regardless of its basic scheduling criteria). Specified in a DO FORCEJOB statement in the rule.

Jobs can be forced for one of the following reasons:

- to start a new process in CONTROL-M (that is, new job submission)
- to enable CONTROL-M to assume full control of an externally submitted job that triggers the event; these jobs are referred to as On Spool jobs, discussed under "On Spool Jobs" on page 683.
- stop the job in which the event occurs

At the end of the current job step, terminate the job in which the event occurred. Specified in a DO STOPJOB statement in the rule.

The following actions can be defined if CONTROL-O is active:

■ invoke a CONTROL-O rule

CONTROL-O rules can be invoked within the current rule. Specified in a DO RULE statement in the rule.

send a message

Messages can be sent to specified locations through the CONTROL-O Shout facility. Specified in a DO SHOUT statement in the rule.

## **CMEM Rule Ordering, Triggering and Deactivation**

CMEM tables, along with their rules, are usually ordered (loaded to memory) when CMEM is started. They can also be refreshed or loaded by an operator command, or manually using the FORCE option in the CMEM Table List screen.

Once a CMEM rule has been loaded in memory, the occurrence of the events specified in its ON statements trigger the rule. All DO statements in the rule are then performed.

More than one rule can be triggered by the occurrence of an event. An event triggers each rule whose ON statement matches the event.

Generally, all actions from all triggered rules are performed.

The one exception occurs when multiple rules are triggered by the same job arrival event and each of the triggered rules contains DO FORCEJOB statements. In this case, the DO FORCEJOB statements of the first triggered rule are performed, but the DO FORCEJOB statements of the other rules triggered by the event are not performed. For more information, see "On Spool Jobs" on page 683.

CMEM rules remain activated, that is, they remain in memory, until they are overridden by the reloading of the rule table or deleted by an operator command.

### **CMEM AutoEdit Variables**

The CMEM facility supports its own set of AutoEdit variables. No other AutoEdit variables can be used by the CMEM facility. Furthermore, in CONTROL-M, these variables can only be specified in CMEM rules, not in job scheduling definitions or JCL.

CMEM AutoEdit variables are resolved upon triggering of the rule. Available CMEM AutoEdit variables are:

Table 218 CMEM AutoEdit Variables (part 1 of 2)

Variable	Description
%%\$Dn	nth qualifier of the data set name. For example, if the data set name is AAA.BBB.CCC, %%\$D2 resolves to BBB. Valid only for rules containing an ON DSNEVENT statement.
%%\$DSN	Name of the data set handled by the rule. Valid only for rules containing an ON DSNEVENT statement.

Table 218 CMEM AutoEdit Variables (part 2 of 2)

Variable	Description
%%\$DSNDISP	Disposition of the data set handled by the rule. Valid only for rules containing an ON DSNEVENT statement.  Possible values are:  C - cataloged  D - deleted  K - kept  N - NOT CATLG2  R - retained  S - scratched  U - uncataloged
%%\$JNAME	Job name. Valid in rules for all types of events.
%%\$SABEND	System abend code of the step whose termination triggered the rule.
%%\$STEPCC	Completion code of the step whose termination triggered the rule.
%%\$UABEND	User abend code of the step whose termination triggered the rule.

## **On Spool Jobs**

On Spool jobs are jobs or started tasks that are submitted externally to CONTROL-M, such as jobs submitted by TSO users or CICS, or jobs received over the NJE network, but are brought under the control of the CONTROL-M monitor using a CMEM rule.

The CMEM rule that causes a job to be an On Spool job, that is, a CMEM rule that brings the external job under the control of the CONTROL-M monitor, must be an ON JOBARRIV rule or CONTROL-O event (ON DSNEVENT or ON STEP) with a DO FORCEJOB statement. To inform CONTROL-M that this is an On Spool job and not a regular FORCEJOB request, the job scheduling definition forced by the DO FORCEJOB must "match" the arriving job, as described below.

CONTROL-M controls the entire life cycle of the job, from determining when to execute the job to performing job post-processing, according to the forced job scheduling definition.

CONTROL-M processes On Spool jobs slightly differently than it processes regular jobs. CONTROL-M does not submit the job because the job has already been submitted. Instead, CONTROL-M releases the job (if held) when the runtime scheduling criteria are met.

Once the job starts execution, whether the job previously required releasing or not, it is controlled by CONTROL-M in the same way that CONTROL-M controls regular jobs. CONTROL-M waits for the job to finish, reads its sysout, and performs all post-processing actions defined in the job scheduling definition.

### **Creating On Spool Jobs**

The following components are necessary to create On Spool jobs:

- 1. job on spool
- 2. CMEM rule
- 3. job scheduling definition

Below is a detailed explanation of what must be defined for each of the above components to create an On Spool job.

### **Job On Spool**

The job must have the following characteristics:

- The job must be submitted with TYPRUN=HOLD to delay its execution and permit CONTROL-M to determine when to run the job.
- The MSGCLASS sysout of the job:
  - **for JES3 users**: must be equal to the CONTROL-M SYSOUT held class
  - for JES2 users: can be any held SYSOUT class

This enables CONTROL-M to read the job sysout and perform post-processing according to the job scheduling definition.

#### **CMEM Rule**

The CMEM rule definition must contain the following:

■ ON JOBARRIV or CONTROL-O Event (ON DSNEVENT or ON STEP) statement

The job name specified in the ON JOBARRIV/ON DSNEVENT/ON STEP statement in this rule must match the name of the job to be monitored. It can be a full job name, or it can be a mask if a group of jobs is to be monitored. For more information, see "ON JOBARRIV: Event Parameter" on page 729, "ON DSNEVENT: Event Parameter" on page 724, or "ON STEP: Event Parameter" on page 733.

#### — NOTE



CONTROL-O users are advised that message rules triggered by \$HASP395 (under JES2) or IEF404I (under JES3) are treated the same as JOBEND rules.

#### DO FORCEJOB statement

The first DO FORCEJOB statement in the rule must force a matching job scheduling definition, described immediately below. For more information, see "DO FORCEJOB: Action Parameter" on page 703.

When monitoring for a group of jobs via job name masking, the JOB parameter in the DO FORCEJOB statement must specify the CMEM AutoEdit variable %%\$JNAME. This allows the CONTROL-M monitor to use the job name mask when performing the job name match process in the job scheduling table. Using this method allows a single CMEM rule to act together with a single job scheduling definition (whose MEMNAME specifies the same job name mask) for jobs that require similar tracking requirements.

## **Job Scheduling Definition**

The job scheduling definition must have the following characteristics:

- The job scheduling definition must be forced by the first DO FORCEJOB statement in the CMEM rule.
- The MEMNAME value in the job scheduling definition must match the name of the external job. A mask can be specified in the MEMNAME field if the same job scheduling definition is used for more than one job.
- Appropriate runtime scheduling criteria for the job must be defined in the job scheduling definition. This enables CONTROL-M to control the execution of the job, that is, when the job must be run.
- Desired post-processing actions must be defined in the job scheduling definition.

## **Handling On Spool Jobs**

On Spool jobs are handled as follows:

■ When the job arrival event occurs, CONTROL-M forces the requested table or job.

If the MEMNAME value in the requested table or job does not match the name of the arriving job, the table or job is forced and processed regularly by CONTROL-M (a job is submitted when its runtime scheduling criteria are met, and so on). If the MEMNAME value in the requested table or job matches the name of the arriving job, the job becomes an On Spool job and CONTROL-M performs the following actions:

- replaces the MEMNAME mask (if a mask was specified in MEMNAME) with the name of the arriving job
- assigns the job ID of the job that triggered the event to the forced job
- forces the job; for details and exceptions see "On Spool Job Scheduling Definition Considerations" on page 686

The forced job appears in the Active Environment screen with status WAIT SCHEDULE ON SPOOL.

- CONTROL-M starts processing the forced job when all runtime scheduling criteria defined in the job scheduling definition are satisfied. If there are no runtime scheduling criteria in the job scheduling definition, CONTROL-M starts processing the job immediately.
- CONTROL-M looks for the job in the spool, to release it, if required.
  - If the external job is waiting for execution in HELD state, that is, if the job arrives on spool with TYPRUN=HOLD, CONTROL-M releases it for execution.
  - Otherwise, CONTROL-M verifies that the job is still in the spool (waiting for execution, executing or ended) before deciding to perform post-processing.
- CONTROL-M waits for the job to finish execution, reads its SYSOUT, analyzes the execution results and performs all the post-processing actions defined in the job scheduling definition.

### NOTE



CONTROL-M can only handle NJE jobs as On Spool jobs when they originate on the same NJE node as that on which CONTROL-M is running.

## On Spool Job Scheduling Definition Considerations

### **Job Forcing Considerations**

Only one On Spool job can be created in response to a job arrival event. However, in several cases, multiple DO FORCEJOB actions might match the arriving job. Each of these cases and the job forcing logic applied to them, to prevent multiple On Spool processes for the same external job, are described below.

■ The job arrival rule contains multiple DO FORCEJOB requests. Each might match the arriving job. In this case, job forcing logic is as follows.

The On Spool process, the match between the external job name and MEMNAME, is performed for the first DO FORCEJOB in the first matching job arrival rule only:

- If a match is found, the job is an On Spool job.
- If a match is not found, the job is not an On Spool job, even if subsequent DO FORCEJOB actions might match.

In either case, all subsequent DO FORCEJOB statements in the same rule (if they exist) are handled normally, that is, not as forcing On Spool jobs.

■ The DO FORCEJOB forces a table in which more than one MEMNAME matches the arriving job. In this case, job forcing logic is as follows.

If a table containing more than one job is forced, by the first DO FORCEJOB statement in the rule, as described above, the first matching job causes the job to be an On Spool job. All the other jobs in the table are forced as regular CONTROL-M jobs, even if they match the job name of the external job.

• Multiple job arrival rules are triggered by the same job arrival event, and each rule contains one or more DO FORCEJOB statements that might match the arriving job. In this case, job forcing logic is as follows.

Only the DO FORCEJOB statements from the first triggered rule are executed, as described above. DO FORCEJOB from all other triggered job arrival rules are ignored.

#### - NOTE



If an On Spool job was purged from the spool but still remains in the Active Jobs file, and another job with the same name arrives on spool and is assigned the same job ID, that later job is not forced.

## **JCL Management Considerations**

When defining JCL, the following issues must be considered:

Any attempt to rerun the job, that is, as a cyclic job, by a DO RERUN statement, or by a manual rerun request, might fail if the JCL of the job is not found in the library specified in the MEMLIB parameter of the job scheduling definition.

- If the job is not submitted with TYPRUN=HOLD, CONTROL-M cannot determine when the job runs, even if runtime scheduling criteria are defined. In this case, the job might start executing before all the runtime scheduling criteria are satisfied. Post-processing, however, is not performed by CONTROL-M until the runtime scheduling criteria are satisfied.
- Since On Spool jobs are not submitted by CONTROL-M:
  - The JCL of the On Spool job cannot contain AutoEdit statements, and SETVAR statements in the job definition are ignored. This is because the job is not submitted by CONTROL-M.
  - Because the job is not submitted by CONTROL-M, the following job scheduling definition parameters are ignored:
    - SCHENV
    - SYSTEM ID
    - NJE NODE
  - NJE enhanced tracking support is inoperative

## **Support for ON DSNEVENT and ON STEP**

A DSNEVENT occurs when a file is deallocated, on the happening of one of the following events:

- the dynamic deallocation of a file
- deallocation of a file on the termination of a job STEP
- deallocation of a file on the termination of a job

A STEP event occurs when a step ends.

To support the ON DSNEVENT and ON STEP parameters, CMEM intercepts the messages written by JES2 or JES3 to JESYSMSG. JESYSMSG is the third SYSOUT of the job.

The DSNEVENT process consists of the following parts:

1. When a job, or STC, or TSO user session starts, the system issues either the IEF403I message or the IEF125I message. CMEM intercepts this message, and checks whether there is at least one ON DSNEVENT or ON STEP rule for the job. If there is at least one rule for the job, CMEM creates the DSNEVENT environment in the address space.

#### NOTE -



A job can only be handled by one CMEM. Therefore, in an environment where more than one CMEM can be active, for example, TEST and production, you must be accurate in defining the ON DSEVENT.

The DSNEVENT environment for an address space remains intact when a new CONTROL-O instance is started while the previous CONTROL-O instance is still active.

- 2. CMEM intercepts allocation, deallocation, and step termination messages, and determines whether there is at least one ON DSEVENT or ON STEP event. If so, CMEM determines
  - whether the DSNEVENT or STEP events fulfil the selection criteria in the rule
  - whether, in the case of a DSNEVENT, the file is a new file or already exists

#### **NOTE**



A DSNEVENT can be triggered only on the CONTROL-O/CMEM that runs on the same z/OS system on which the event occurs. Even a focal CONTROL-O or CMEM in a Sysplex environment is insufficient. This is because CONTROL-O or CMEM intercepts the messages written to JESYSMSG while they are being written, which can be done only on the same system where the event happens.

- 3. CMEM triggers the rule according to the following principles:
  - A. An ON STEP rule is always triggered when a STEP ends. However, a STEP is ignored when a rule is not executed due to one of the following:
    - a JCL error
    - failure to encounter a previous step condition code
  - B. An ON DSNEVENT can occur when a deallocation message is written, or when a step terminates. Whether it occurs depends on
    - how the file is deallocated
    - the setting of the STEPRC field in the DSNEVENT criteria, as follows:
    - If the value of STEPRC is null, the rule is triggered at the time of deallocation.
    - If the value of STEPRC is other than null, the rule is rechecked at the termination of the step to ascertain the STEPRC criteria.

#### C. Deallocation occurs when

- the file is dynamically deallocated
- the step ends, in the case of all allocated files in this step, whether the files were allocated by JCL or were dynamically allocated
- the job terminates, in the case of any file that was not released on step termination, for example, if DISP is set to PASS

For CMEM actions to be triggered in this way, the following conditions must be satisfied:

- Either the IEF403I message or the IEF125I message must appear in the job log.
- The job, STC, or TSO user session must have its MSGLEVEL set to (x,1), to ensure that allocation, deallocation and step termination messages are written to the JESYSMSG. It is not sufficient for these messages to appear only in the system log or job log.
- At least one ON DSNEVENT or ON STEP rule must be ordered and ready before the job, STC, or TSO user session starts.

#### NOTE -



If the value of ON DSNEVENT is * (asterisk), every job, STC, or TSO user session is within the DSNEVENT environment. BMC Software recommends that you do not use this type of DSNEVENT because of

- the overhead that results from CMEM analyzing every message written to the JESYSMSG
- the CMEM limitation that only one CMEM can handle a DSNEVENT for a job In an environment where more than one CMEM is active on the same system, this definition may cause the wrong CMEM to trap the event.

## **Regular Allocation and Deallocation**

In the case of regular files, meaning files not managed by SMS, CMEM traps the following deallocation messages:

- IEF283I
- IEF285I
- IEF287I

## **SMS Support**

In the case of SMS-managed data sets, CMEM traps the IGD101I and IGD104I allocation and deallocation messages, and determines whether the file is new according to the following rules:

- If both messages are issued, the file is new.
- If the IGD101I message is not found, CMEM treats the file as not new.

## **Generation Data Sets (GDG)**

For CMEM to support Generation Data Sets (GDG), the following messages must be found:

- IGD105I
- IGD107I
- IGD108I
- IGD17101

#### NOTE



The messages that are required by CMEM for the purposes of this and the preceding sections are liable to be changed as a result of IBM changes in data set processing.

## CMEM Support for FTP

CMEM actions can be triggered by the transfer of files by FTP products.

In order to enable CMEM rules to be triggered by such file transfers, do the following:

- 1. Use one of the following methods to insert the expression MSGLEVEL=(1,1) in the STC of the FTP product:
  - Customize the JESxPARM member in the SYS1.PARMLIB library.
  - Modify the job statement in the PROCLIB library member that contains the procedure JCL for the STC of the FTP product.
- 2. Start the CMEM monitor with at least one DSNEVENT rule that refers to the STC of the FTP product. You can use a dummy DSNEVENT rule that forces CMEM to monitor the STC of the FTP product.
- 3. Start the FTP product.

4. Modify existing rules, or order new rules (or do both). Rules that have been modified or ordered before a file reaches the FTP server are applied to all files subsequently transferred by the FTP product.

CMEM triggers rules when the requirements of an ON DSNEVENT statement are satisfied. If an FTP product fails to transmit a data set and issues a message relating to this failure, CMEM cannot react to that message. However, you can use CMEM rules to react to FTP product messages.

### - NOTE -



Do not use the STEPRC subparameter in conjunction with the FTP* setting for ON DSNEVENT, because the same FTP procedure can serve many requests before being terminated.

## **CMEM Support for IBM FTP**

The IBM FTP process is executed under the OMVS address space, which is BPXAS. BPXAS address spaces do not usually write messages to the JESYSMSG.

In order to enable CMEM DSNEVENT support for IBM FTP, the following occurs:

1. When the CMEM monitor starts, it activates the OpenEdition interface for processes in the BPXAS. Having done this, CMEM issues the following messages:

CTO782I SUBSYSTEM REGISTERED WITH OPENEDITION INTERFACE
CTO783I INITIALIZATION OF OPENEDITION ENVIRONMENT ENDED
SUCCESSFULLY

When CMEM initialization is complete, message CTO147I is displayed. After this, CMEM gets control every time that an OpenEdition process starts in the BPXAS address space.

- 2. When a new process starts in the BPXAS, the interface routine issues the CTO403I pseudo-message. This pseudo-message causes CMEM to simulate IEF403I processing, which is described in "Support for ON DSNEVENT and ON STEP" on page 688.
- 3. Under z/OS, IBM FTP functions as a UNIX process running under z/OS. Therefore it follows the UNIX standard, one aspect of which is that the name of the job is set to the user ID of the person who issued the FTP request. This UNIX standard creates a problem, because the operator who writes the CMEM rule must know at that stage which user will issue an FTP request.

The solution to this problem is to use the statement ON DSNEVENT FTP*. The result is that any FTP process will trigger the rule.

### NOTE -



Do not use the STEPRC subparameter in conjunction with the FTP* setting for ON DSNEVENT, because the same FTP procedure can serve many requests before being terminated.

# **Rule Parameters – Summary**

Figure 322 CMEM Rule Definition Screen

RL: JOBNAM1 LIB CTM.PROD.RULES COMMAND ===>		TABLE: CMEMRULE SCROLL===> CRSR
ON JOBARRIV = JOBNAM1 JTYPE SMFID OWNER CTMCTLM GROUP THRESHOLD DESCRIPTION CONVERSION: ON JOB JOBNAM1 DESCRIPTION	SYSTEM MODE PROD	And/Or/Not RUNTSEC NONE
DO FORCEJOB = TABLE TABLE1 JOB LIBRARY CTM.PROD.SCHEDULE DO		DATE ODAT
====== >>>>>>>>>> END OF RULE DEFINI	TION PARAMETERS <<	
FILL IN RULE DEFINITION. CMDS: CAPS. EDIT	, SHPF	21.00.36

The parameters of the CMEM Rule Definition screen are divided into the following categories:

- **■** Event Selection Parameters
- General Parameters
- Action Parameters

A brief summary of the parameters in each category is provided on the following pages. This is followed by a detailed description of each parameter in alphabetical order.

## **Event Selection Parameters**

The following parameters identify the events that trigger the rule.

**Table 219 CMEM Event Selection Parameters** 

Parameter	Description
ON statement	<ul> <li>Event criteria that must be satisfied for the rule to be triggered.</li> <li>Subparameters may be displayed. Valid ON statements are:</li> <li>ON DSNEVENT – name (or mask) of the job to be monitored for data set or NCT2 events</li> <li>ON JOBARRIV – job name (or mask) of a job or started task that arrived on the JES spool from any source</li> <li>ON JOBEND – job name (or mask) of a job or started task that terminated</li> <li>ON STEP – job step whose termination is to be monitored for a specified return code or status</li> </ul>

Subparameters of these parameters are described in the detailed description of each parameter later in this chapter.

## **General Parameters**

The following parameters contain general information about the rule.

**Table 220 CMEM General Parameters** 

Parameter	Description
OWNER	ID of user who requests CMEM services
GROUP	Name of a group of rules
MODE	CMEM rule operation mode
RUNTSEC	Type of runtime security checks to be performed for the rule
DESCRIPTION	Free-text description of the rule definition

# **Action Parameters**

The following parameters (DO statements) specify actions to be performed.

**Table 221 CMEM Action Parameters** 

Parameter	Description
DO statement	Action to be performed when the rule is triggered. Subparameters may be displayed. Valid DO statements are:  ■ DO COND – add or delete a prerequisite condition  ■ DO FORCEJOB – force a job order under CONTROL-M  ■ DO STOPJOB – stop execution of the remaining steps of the job that triggered the rule  The following actions can be defined if CONTROL-O is active:  ■ DO RULE – invoke a CONTROL-O rule from within the current rule  ■ DO SHOUT – issue a message to a specified destination using the Shout facility

# **Parameter Descriptions**

The following pages contain detailed descriptions of all parameters available in the CMEM Rule Definition screen. Parameters are arranged in alphabetical order. Within each parameter, subparameters are arranged according to the order of the fields on the screen.

Each parameter begins on a new page, including:

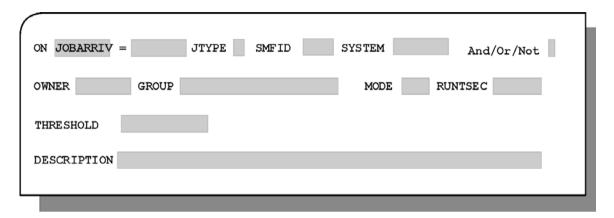
- a brief explanation of the purpose of the parameter
- the format required for defining the parameter within an extract of the CMEM screen
- general information explaining the parameter and its usage
- where applicable, some practical examples illustrating implementation of the parameter

For more information on the CMEM Rule Definition facility, see Chapter 2, "Online Facilities."

# **DESCRIPTION: General Parameter**

Description of the rule to be displayed in the Rule List screen.

Figure 323 DESCRIPTION Parameter Format



Optional. The DESCRIPTION parameter consists of one or more lines that can contain free text. Each line is 61 characters in length. Upon typing text in a DESCRIPTION line and pressing **Enter**, a new DESCRIPTION line is opened.

### **General Information**

The DESCRIPTION parameter does not affect rule processing. The text entered in the first DESCRIPTION line appears to the right of the rule name in the Rule List screen. It is intended to let the user know at a glance the purpose of, or some other key information about, the rule. The text can be typed in any language.

## **Example**

The description START THE BATCH SHIFT appears next to the rule name in the Rule List screen.

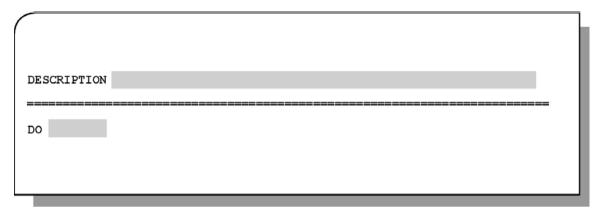
## Figure 324 DESCRIPTION Parameter Example

RL: CICSPROD LIB CTM.PROD.RULES COMMAND ===>	TABLE: STCS SCROLL===> CRSR
ON JOBEND = CICSPROD JTYPE SMFID SYSTEM OWNER ADMIN GROUP CICS MODE PROD THRESHOLD DESCRIPTION START THE BATCH SHIFT DESCRIPTION	And/Or/Not
/* INFORM CONTROL-M THAT THE BATCH CAN BE STARTED DO COND = START-BATCH ODAT + DO	
>>>>>>> END OF RULE DEFINITION PARAMETERS	 <<<<<< <del></del>
FILL IN RULE DEFINITION. CMDS: CAPS, EDIT, SHPF	21.00.36

# **DO statement: Action Parameter**

Actions to perform when the rule is triggered.

Figure 325 DO Parameter Format



At least one DO statement must be specified in each rule. Specify DO statements as follows:

- Type the action keyword (such as COND) in the DO field and press **Enter**.
- When required, subparameter fields are displayed. Fill in the subparameters and press Enter again.

Multiple DO statements can be specified. After entering a DO statement, another DO line is automatically displayed. Multiple DO statements have an AND relationship and are performed sequentially.

The following are valid DO actions. Each is discussed individually in this chapter.

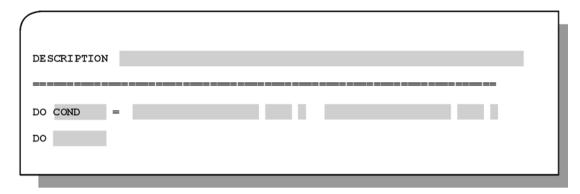
**Table 222 DO Parameter Actions** 

Action	Description	
DO COND	Adds and/or deletes one or more prerequisite conditions	
DO FORCEJOB	Forces a job	
DO STOPJOB	Stops execution of the job that triggered the rule, at the end of the current step	
If CONTROL-O is active:		
DO RULE	Invokes a CONTROL-O rule	
DO SHOUT	Sends a message to a specified destination	

# **DO COND: Action Parameter**

Add or delete prerequisite conditions.

Figure 326 DO COND Parameter Format



Optional. Type the word COND (or its abbreviation CON) in the DO field and press **Enter**. The following subparameters are displayed:

Table 223 DO COND Subparameters (part 1 of 2)

Subparameter	Description	
condition	User-supplied, descriptive name of 1 through 20 characters used to identify the condition. Only trailing blanks are permitted.	
dateref	4-character date reference associated with the condition. Valid values are:	
	<ul> <li>date – specific date, in mmdd or ddmm format, depending on the site standard</li> <li>ODAT – resolves to the current installation working date Default.</li> <li>DATE – resolves to the current system date</li> <li>STAT – static Indicates that the condition, such as IMS-ACTIVE, is not date-dependent</li> </ul>	
Note: Before STAT was introduced, date 0101 was reco to be used in conditions that were not date-dependen 0101, STAT is not a date, and it operates differently. A STAT when defining conditions that are not date-dep		
	**** /\$\$\$\$ – all dates Valid only for deleting prerequisite conditions. Either value (**** or \$\$\$\$) results in the deletion of all matching prerequisite conditions regardless of date.	

Table 223 DO COND Subparameters (part 2 of 2)

Subparameter	Description
cond_opt	Indicator of whether to add or delete the prerequisite condition. Valid values are:
	<ul> <li>+ - add the prerequisite condition</li> <li> delete the prerequisite condition</li> </ul>

### **General Information**

When a rule containing a DO COND statement is triggered, the designated prerequisite conditions are added or deleted (as specified) from the IOA Conditions file by the CONTROL-M monitor.

A prerequisite condition can define any user-specified situation. The following are a few examples of prerequisite conditions:

IMS-ACTIVE		
WEEKEND		
SALARY-OK		

Prerequisite conditions created or deleted by the DO COND parameter can activate or deactivate CONTROL-O rules, or trigger (or stop) the execution of processes (jobs, and so on) in CONTROL-M, CONTROL-D and other environments.

CMEM AutoEdit System variable %%\$JNAME and, for ON DSNEVENT events, variables %%\$Dx, %%\$DSN, or %%\$DSNDISP can be specified in condition names in DO COND statements and are replaced (resolved) at time of rule triggering. For more information, see "CMEM AutoEdit Variables" on page 682.

Representative dates (such as ODAT) are resolved to the actual corresponding date in the site-standard format.





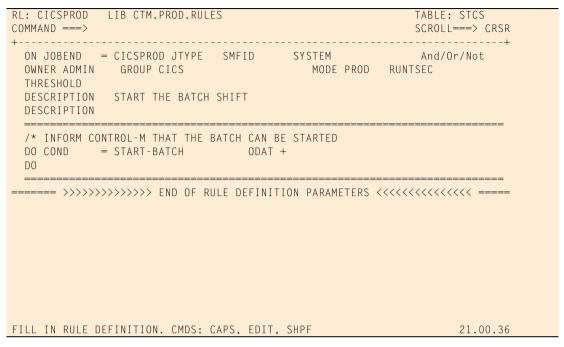
Long condition names cannot be used in CMEM rule definitions.

## **Example**

### **Example 1**

When job CICSPROD ends, this rule sets the condition necessary for the batch shift to begin.

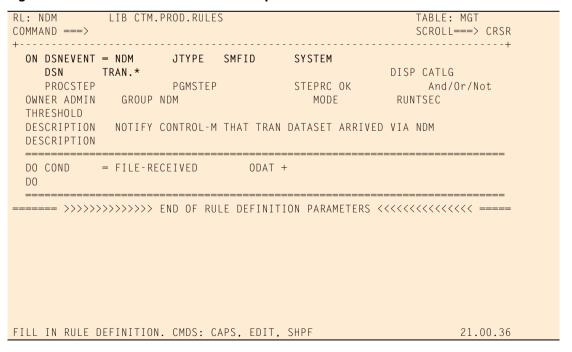
Figure 327 DO COND Parameter – Example 1



### **Example 2**

When a data set with name prefix TRAN arrives by file transfer product CONNECT DIRECT (formerly called NDM), add prerequisite condition FILE-RECEIVED to notify CONTROL-M that the data set was received.

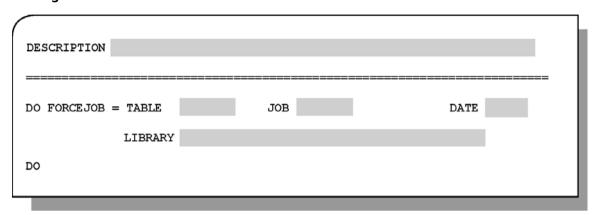
Figure 328 DO COND Parameter – Example 2



# **DO FORCEJOB: Action Parameter**

Force a job.

Figure 329 DO FORCEJOB Parameter Format



Optional. Type the word FORCEJOB (or its abbreviation F) in the DO field and press **Enter**. The following subparameters are displayed:

**Table 224 DO FORCEJOB Subparameters** 

Subparameter	Description
TABLE	Name of a scheduling table, up to eight characters. Mandatory.
JOB	Name of the job to be triggered. Optional. If blank, all jobs in the table are forced.
	If AutoEdit System variable %%\$JNAME is specified, it resolves to the name of the job that triggered the rule.
DATE	<ul> <li>Scheduling date of the job. Valid values are:</li> <li>date – specific 6-digit date, in format mmddyy, ddmmyy, or yymmdd, depending on the site standard</li> <li>ODAT – resolves to the current installation working date Default.</li> <li>DATE – resolves to the current system date</li> </ul>
LIBRARY	Name of the scheduling library containing the specified table. Mandatory.

### **General Information**

DO FORCEJOB places a job order in the CONTROL-M Active Jobs file, even if the basic scheduling criteria of the job are not satisfied. For more information, see "Job Ordering and Job Forcing" on page 66.

The DO FORCEJOB statement in CMEM enables CONTROL-M to order CONTROL-M scheduling tables based on the occurrence of an event, for example, job arrival, job end, data set event, or step event.

The DO FORCEJOB statement is executed by the CONTROL-M monitor.

If the CONTROL-M monitor is not active, the DO FORCEJOB request is queued and performed when the CONTROL-M monitor becomes active.

DO FORCEJOB logic works differently for job arrival events than for job end, data set or step events:

#### **Job End Events**

DO FORCEJOB statements specified in a job end event rule are performed only if the terminating job is not under CONTROL-M.

### **Data set or Step Events**

Data set or step event rules are performed regardless of where the job was submitted. However, if the triggering job was ordered or submitted by CONTROL-M, the job will not become an On Spool job. For more information about On Spool jobs, see "On Spool Jobs" on page 683.

#### **Job Arrival Events**

For the first DO FORCEJOB statement in a rule:

- If the job that triggered the job arrival event was submitted by CONTROL-M, the DO FORCEJOB statement is ignored.
- If the job that triggered the job arrival event was not submitted by CONTROL-M, CONTROL-M forces the requested table or job. CONTROL-M scans the forced table looking for a job whose MEMNAME value matches, or is a mask for, the name of the job whose arrival triggered the rule.
  - If a matching job is found, it becomes an On Spool job. For more information, see "Creating On Spool Jobs" on page 684.
  - If a matching job is not found, or more than one job is ordered, all other jobs are not On Spool jobs, and are processed normally by CONTROL-M.

For other DO FORCEJOB statements in the same rule:

- DO FORCEJOB is performed regardless of the source of the job.
- The table is forced.

DO FORCEJOB is not executed if a preceding ON JOBARRIV rule with a DO FORCEJOB action was already executed for this event.

### — NOTE –



When a DO FORCEJOB request fails because the scheduling table is in use, CONTROL-M may try again to execute the job, depending on the values set for the FORCE#RT and FORCE#WI installation parameters. For more information on the FORCE#RT and FORCE#WI installation parameters, see the customization chapter of the *INCONTROL* for z/OS Installation Guide.

## **Examples**

### **Example 1**

Control all jobs not submitted by CONTROL-M.

### Figure 330 DO FORCEJOB – Example 1

```
LIB CTM.PROD.RULES
                                            TABLE: JOBS
COMMAND ===>
                                            SCROLL===> CRSR
 ON JOBARRIV = * JTYPE J SMFID SYSTEM
                                             And/Or/Not
 OWNER ADMIN GROUP EXTJOBS
                               MODE
                                        RUNTSEC
 THRESHOLD
 DESCRIPTION CONTROL ALL JOBS NOT SUBMITTED BY CONTROL-M
 DESCRIPTION
 DO FORCEJOB = TABLE ANYJOB JOB
          LIBRARY CTM.PROD.SCHEDULE
 D0
 FILL IN RULE DEFINITION. CMDS: CAPS, EDIT, SHPF
                                                 21.00.36
```

Scheduling table ANYJOB must contain at least one job scheduling definition.

### **Example 2**

Control all jobs submitted by CICS. These fall into the following groups: Jobs whose name starts with A and jobs whose name starts with C.

Figure 331 DO FORCEJOB – Example 2

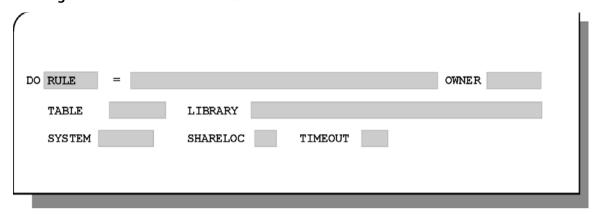
Scheduling table CICSJOB must contain at least two job scheduling definitions:

- One must contain a MEMNAME value beginning with A.
- Another must contain a MEMNAME value beginning with B.

# **DO RULE: Action Parameter**

Invoke a CONTROL-O rule from within the current rule. Available only if CONTROL-O is active.

Figure 332 DO RULE Parameter Format



Optional. Type the word RULE (or its abbreviation RU) in the DO field and press **Enter**. The following subparameters are displayed:

**Table 225 DO RULE Subparameters** 

Subparameter	Description
rulename	Name of the CONTROL-O rule to be executed. A maximum of eight characters can be entered.
	<b>Note:</b> The rule to be executed must contain an ON RULE statement with the same rule name specified in this parameter.
args	Optional input and output arguments to be passed to the rule can be specified, following the rulename and separated from it by a blank. Arguments must be valid AutoEdit expressions separated by commas. An unlimited number of arguments can be specified. However, the combined length of the rulename and arguments cannot exceed 45 characters.
OWNER	Value of the OWNER field in the invoked rule. This subparameter is used for security purposes. Optional.
TABLE	Name of the CONTROL-O rule table in which the invoked rule resides. When ALL is entered, it implies that the invoked rule may reside in any rule table. If a table name is not entered, the current rule table is assumed by default.
LIBRARY	Name of the CONTROL-O rule table library where the invoked rule resides. When ALL is entered, it implies that the invoked rule may reside in any rule table library. If a library name is not specified, the current rule table library is assumed by default.

### **General Information**

To define a DO RULE statement in a CMEM rule, and to access a CMEM rule containing a DO RULE statement, CONTROL-O must be installed.

#### - NOTE



To order a CMEM rule containing a DO RULE statement and to invoke the CONTROL-O rule specified in the CMEM DO RULE statement, CONTROL-O must be active.

When a DO RULE statement is encountered during rule processing, CONTROL-O invokes the specified rule. When processing of the invoked rule is completed, processing continues sequentially from the point after the DO RULE statement in the initial (calling) rule.

When a DO RULE statement is executed, the specified rule is searched for among the loaded rules according to the specified rule name, table, library, and owner. If the rule is found but is not active, for example, if the runtime scheduling criteria are not satisfied, the "invoked" rule is not executed and the calling rule continues execution with the next DO statement.

The CMEM calling rule can pass an argument string as input to the called rule. This argument string can contain CMEM AutoEdit expressions that are resolved at time of rule execution. The argument string can be accessed by the called rule through CONTROL-O System variable %%\$ARGS. If a called rule calls another CONTROL-O rule, the %%\$ARGS values passed in the earlier call are overwritten by the %%\$ARGS values passed by the later call. For information about the AutoEdit facility in CONTROL-O, see the *CONTROL-O User Guide*.

A CONTROL-O rule specified with an ON RULE statement can be invoked any number of times by DO RULE calls.

A called CONTROL-O rule can invoke other CONTROL-O rules using DO RULE statements. Nesting of DO RULE calls, for example, rule 1 calls rule 2, that calls rule 3, up to 20 deep is supported. A CONTROL-O rule can be called recursively.

## **Example**

When a data set named PROD.TRAN.* is cataloged by TCPIP, invoke a CONTROL-O rule that starts a task to process it.

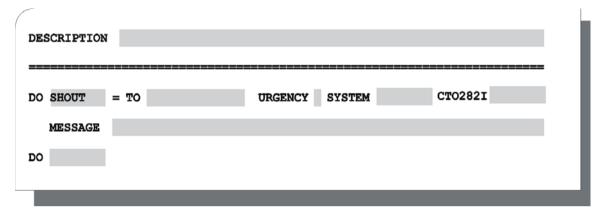
Figure 333 DO RULE Example

```
RL: TCPIP LIB CTM.PROD.RULES
                                                    TABLE: TRANS
COMMAND ===>
                                                    SCROLL===> CRSR
+-----
  ON DSNEVENT = TCPIP JTYPE SMFID SYSTEM
    DSN PROD.TRAN.*
PROCSTEP PGMSTEP STEPRC OK And/Or/Not RUNTSEC
    DSN PROD.TRAN.*
  OWNER ADMIN GROUP TCPIP
  THRESHOLD
  DESCRIPTION WHEN DATASET PROD.TRAN.* IS CATALOGED BY TCIP.
  DESCRIPTION START A TASK TO PROCESS IT
  DESCRIPTION
  /* START A STARTED TASK TO PROCESS THE RECEIVED FILE.
  /* WHEN THE DATASET IS CATALOGED, INVOKE RULE PROCFILE.
  /* PARAMETERS PASSED ARE THE STC NAME AND THE TIMEOUT VALUE.
  DO RULE = PROCFILE %%$DSN
                                                   OWNER PROD
    TABLE PRODRULE LIBRARY CTO.PROD.RULES
                   SHARELOC
    SYSTEM
  D0
===== >>>>>>>> END OF RULE DEFINITION PARAMETERS <<<<<<<<< =====
FILL IN RULE DEFINITION. CMDS: CAPS, EDIT, SHPF,
                                                     21.00.36
```

# **DO SHOUT: Action Parameter**

Send ("shout") a message to a specific destination. Available only if CONTROL-O is active.

Figure 334 DO SHOUT Parameter Format



Optional. Type SHOUT in the DO field and press **Enter**. The following subparameters are displayed:

Table 226 DO SHOUT Subparameters (part 1 of 3)

Subparameter	Description
TO	<ul> <li>Destination of the message (1 through 16 characters). Mandatory. Valid values are:</li> <li>■ U-userid or USERID-userid – Writes the message to the IOA Log file under the specified user ID. userid must be 1 through 8 characters.</li> <li>■ OPER [dd] [-{rrr   -ccc}] – Sends the message to operator consoles, according to the optional subparameters dd, rrr, and ccc — dd — Descriptor code (from 0 through 16). For more detailed information regarding descriptor codes, refer to the IBM publication Routing and Descriptor Codes, GC38-1102.</li> <li>— rrr — Route code (from 0 through 128). For more detailed information regarding route codes, refer to the IBM publication Routing and Descriptor Codes, GC38-1102. Route code and console ID are mutually exclusive.</li> <li>— -ccc — Console ID number (preceded by a hyphen) of the console to which the message is to be shouted. Console ID and route code are mutually exclusive.</li> </ul>

Table 226 DO SHOUT Subparameters (part 2 of 3)

Subparameter	Description
	TSO - loginid [;Nn   ;Mm   ;NnMm   ;Lname] – Sends the message to the user identified by the specified logon ID. logonid is mandatory (1 through 7 characters).  An optional second value, indicating the computer and/or node (such as Nn) of the TSO logonid, can be entered, as follows:  Under JES2:  Valid values are: Mm, Nn or NnMm, where:  ■ m is the machine ID (the computer in JES2, not the 4-character SMF ID). For more information, see the description of specifying IOA CPU in the discussion of the customization process in the INCONTROL for z/OS Installation Guide.  ■ n is the 1 or 2 character JES/NJE node ID.  Under JES3:  The only valid value is Lname, where name is the logical JES name of the machine (that is, the name as used in JES3, command *T, not the SMF system ID).
	For more information, see the description of specifying IOA CPU in the discussion of the customization process in the <i>INCONTROL</i> for <i>z/OS Installation Guide</i> .  Note: A shout to a TSO user performs a TSO SEND command that
	<ul> <li>may require authorization at the receiving node.</li> <li>U-M: mail-name-prefix – Sends a message by mail to the recipient identified by mail-name-prefix (1 through 12 characters).</li> <li>U-ECS – Sends messages to the CONTROL-M/Enterprise Manager user. For more information on this feature, see the section on shouting to CONTROL-M/Enterprise Manager in Chapter 3, "Job Production Parameters."</li> </ul>
URGENCY	Determines the priority level of the message. For more information, see "The URGENCY subparameter" on page 713. Valid values are:  R - Regular. Default. U - Urgent. V - Very urgent.
SYSTEM	Name of the system (computer) where the message must be directed. A name of one to eight alphanumeric characters can be entered. Mask characters (* and ?) are supported for this subparameter.
	Note: If no SYSTEM value is specified, the message is sent to the system identified by reserved user-defined variable %%\$COMMSYS in a preceding DO SET statement. For a description of %%\$COMMSYS, see the CONTROL-O User Guide. If %%\$COMMSYS is not specified, the message is issued on the current system.  Can be used only when CONTROL-O is installed.
CTO282I	Indicates if the message ID is prefixed by CTO282I. Optional. Valid values are:  ■ Y (Yes) – The message ID is prefixed by CTO282I. Default. ■ N (No) – The message ID is the first word of the message text.

Table 226 DO SHOUT Subparameters (part 3 of 3)

Subparameter	Description
MESSAGE	Message text. Maximum Length: 60 characters. Mandatory.

### **General Information**

The message is sent to the required destination when the accompanying ON statement criteria are satisfied.

It is also possible to shout to a ROSCOE user. For additional information, see your INCONTROL administrator.

### **Subparameter TO**

Type TO=USERID-*userid* to write the message to the IOA Log under the user ID specified in the parameter.

Type TO=OPER[dd]-{rrr,-ccc} to send the message to all operator consoles, or to operator consoles selected according to route code (rrr) or console ID number (-ccc). The descriptor code (dd) determines the type of message displayed. The dd, rrr, and -ccc parameters are optional and can be assigned any valid value. Dashes (-) are used to separate the parameters specified.

For more detailed information regarding route and descriptor codes, refer to the IBM publication *Routing and Descriptor Codes, GC38-1102*.

### **Examples**

**Table 227 DO SHOUT OPER Subparameter Examples** 

Subparameter	Description
OPER	Send the message to all operator consoles.
OPER2	Send a highlighted unrollable message (descriptor code 2) to all operator consoles.
OPER-5	Send a message to operator consoles associated with route code 5.
OPER2-5	Send a highlighted unrollable message to operator consoles associated with route code 5.
OPER-4	Send a message to operator console ID 04.
OPER2-4	Send a highlighted unrollable message (descriptor code 2) to operator console ID 04.

Type TO=TSO-*logonid* to send the message to a groupid or logonid. The Shout facility first searches the IOA Dynamic Destination table for the specified ID. If the table contains an entry that matches the value, the content of the entry is used as the target for the shouted message. The entire TO field is used. Therefore, when directing the message to a remote user, do not append Nn or Mm. Instead, do this in the IOA Dynamic Destination Table itself. For more information, see the description of Dynamic Destination Tables in the *INCONTROL* for *z/OS* Administrator Guide.

If no matching ID is found in the Dynamic Destination table, the Shout facility assumes the specified ID is a logonid. It then creates a TSO message that it hands over to MVS. MVS then sends the message to that logonid. If the logonid does not exist, MVS cannot send the message, but no error message is generated. When a second value is used, the message is sent to the TSO logonid in the specified computer or node (machine ID). To determine the machine ID under JES2, enter JES command SD MEMBER.

### The URGENCY subparameter

The URGENCY value indicates the urgency level of the message.

In addition, if the destination is USERID-*userid* (or U-*userid*), the user can control, according to urgency, which messages are displayed when the IOA Log file is accessed. Urgent and very urgent messages are highlighted on the screen. For more details, see "IOA Log Facility" on page 296

### The CTO282I Subparameter

By default, the CTO282I subparameter has a value of Y, and CTO282I is placed as the message ID preceding the message text. When CTO282I is set to N, the first word of the message text becomes the message ID.

#### **CONTROL-O AutoEdit Variables**

CONTROL-O AutoEdit variables embedded in the TO and MSG subparameters are automatically resolved at time of rule activation. For more information about the AutoEdit facility, see Chapter 5, "JCL and AutoEdit Facility,".

## **Example**

When started task DB2MSTR ends, issue a message to the DBA who is on duty. Notice the use of the generic TSO user name, that the Dynamic Destination table interprets to be one or more TSO users.

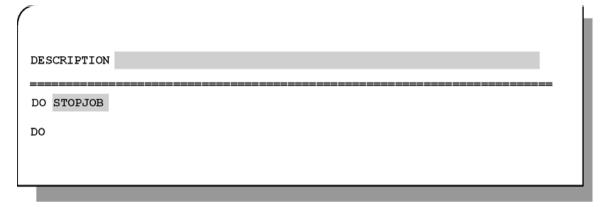
Figure 335 DO SHOUT Parameter Example

RL: DB2MSTR LIB CTM.PROD.RULES COMMAND ===>		TABLE: STCS SCROLL===> CRSR
ON JOBEND = DB2MSTR JTYPE S SMFID OWNER ADMIN GROUP DB2 THRESHOLD DESCRIPTION WARN DBA THAT DB2 MASTER END DESCRIPTION	SYSTEM MODE RUI	And/Or/Not NTSEC
DO SHOUT = TO TSO-DBA URGENCY MESSAGE DB2 MASTER ENDED - PLEASE CHECK DO		CT0282I
>>>>>>>> END OF RULE DEFINITIO	N PARAMETERS <<<<	
FILL IN RULE DEFINITION. CMDS: CAPS, EDIT, S	HPF	21.00.36

# **DO STOPJOB: Action Parameter**

Stop execution of the job that triggered the rule after the current step.

Figure 336 DO STOPJOB Parameter Format



Optional. Type STOPJOB, or its abbreviation ST, in the DO field and press Enter.

### **General Information**

When DO STOPJOB is performed, the job that triggered the rule is terminated after the current step, and no further steps (including those marked COND=EVEN or COND=ONLY) are executed. An appropriate message is written to the job log. If the stopped job is controlled by CONTROL-M, it terminates with a status of ENDED NOTOK.

DO STOPJOB is not executed for TSO users.

DO STOPJOB is meaningful only:

- for data set events or step events
- when there are additional steps in a job or started task

## **Example**

If the production data set disposition is NOT CATLGD 2, stop the job.

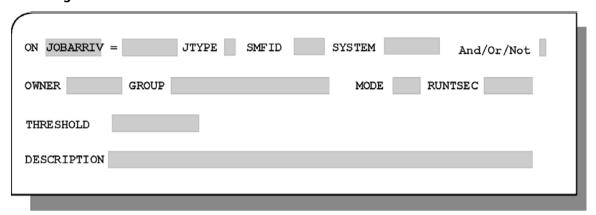
Figure 337 DO STOPJOB Parameter Example

```
RL: PROD*
            LIB CTM.PROD.RULES
                                                      TABLE: JOB
COMMAND ===>
                                                     SCROLL===> CRSR
+-----
 ON DSNEVENT = PROD* JTYPE J SMFID SYSTEM
    DSN PROD.*
                                                 DISP NCT2
                   PGMSTEP STEPRC PRODJOBS MODE
    PROCSTEP
                                                  And/Or/Not
                                     MODE RUNTSEC
  OWNER ADMIN GROUP PRODJOBS
  THRESHOLD
  DESCRIPTION STOP THE JOB ON NCT2 DISPOSITION
  DESCRIPTION
  /* STOP THE JOB
 DO STOPJOB
  DO
 ----- >>>>>>>> END OF RULE DEFINITION PARAMETERS <<<<<<<<< ----
FILL IN RULE DEFINITION. CMDS: CAPS, EDIT, SHPF
                                                           21.00.36
```

# **GROUP: General Parameter**

Name of the group to which the rule belongs.

Figure 338 GROUP Parameter Format



Optional. Name of 1 through 20 characters, with no embedded blanks.

## **General Information**

The GROUP parameter is used to provide more convenient handling of rules. It enables retrieval of information on a group basis. The group name appears in all important IOA Log file messages relating to the rules of the group.

## **Example**

The rule that instructs CONTROL-M to start the batch shift when CICSPROD ends belongs to group CICS.

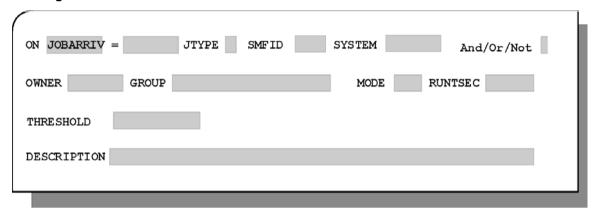
Figure 339 GROUP Parameter Example

RL: CICSPROD LIB CTM.PROD.RULES COMMAND ===>	TABLE: STCS SCROLL===> CRSR
ON JOBEND = CICSPROD JTYPE SMFID SYSTEM OWNER ADMIN GROUP CICS MODE PROD THRESHOLD DESCRIPTION START THE BATCH SHIFT DESCRIPTION	And/Or/Not
/* INFORM CONTROL-M THAT THE BATCH CAN BE STARTED DO COND = START-BATCH ODAT + DO	
>>>>>>>>> END OF RULE DEFINITION PARAMETERS	 5
FILL IN RULE DEFINITION. CMDS: CAPS, EDIT, SHPF	21.00.36

# **MODE: General Parameter**

Rule operation mode.

Figure 340 MODE Parameter Format



Optional. Valid values, and their abbreviations, for the MODE parameter are:

**Table 228 MODE Parameter Values** 

Value	Description
PROD (P)	Standard Production mode. The rule is processed normally. Default.
TEST (T)	Test mode. Actions are not performed, but are written to a test journal.
LOG (L)	Log mode. The rule is processed normally and all identified events and actions are written to a test journal.

## **General Information**

Test mode provides the opportunity to test the effects of a rule definition without actually performing the specified DO actions.

Log mode provides a transition between Test and Production mode. Like Production mode, Log mode enables performance of the specified DO actions. However, Log mode also records the trace information in the test journal for tracking purposes.

When tracking of the performed actions for test purposes is no longer required, the rule can be placed in Production mode.

For sites in which CONTROL-O is not installed, or in which the CONTROL-O Automation Log facility is not active, the trace information is written to the sysout referenced by DD statement DAACTLOG.

For CONTROL-O sites in which the Automation Log facility is active, the trace information is recorded in the Automation log. For more information, see the *CONTROL-O User Guide*.

## **Example**

The rule that instructs CONTROL-M to start the batch shift when CICSPROD ends is activated in Production mode.

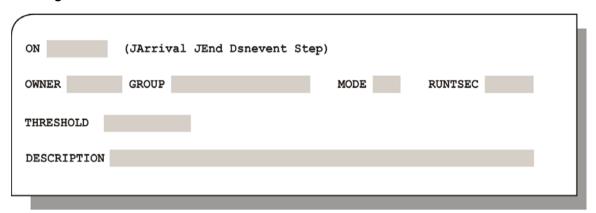
Figure 341 MODE Parameter Example

RL: CICSPROD LIB CTM.PROD.RULES COMMAND ===>	TABLE: STCS SCROLL===> CRSR
ON JOBEND = CICSPROD JTYPE SMFID SYSTEM OWNER ADMIN GROUP CICS MODE PROD DESCRIPTION START THE BATCH SHIFT DESCRIPTION	And/Or/Not
/* INFORM CONTROL-M THAT THE BATCH CAN BE STARTED DO COND = START-BATCH ODAT + DO	
>>>>>>>> END OF RULE DEFINITION PARAMETERS	······································
FILL IN RULE DEFINITION. CMDS: CAPS, EDIT, SHPF	21.00.36

## **ON statement: Event Parameter**

Event selection criteria that trigger performance of the rule.

Figure 342 ON Parameter Format



Mandatory. At least one ON statement must be entered. Type an ON statement, or its abbreviation, in the ON field and press **Enter**. Additional subparameters are displayed.

The following are valid ON statements (and their abbreviations). Each is described in detail later in this chapter.

**Table 229 ON Parameter Statements** 

Statement	Description
ON DSNEVENT (D)	Name (or mask) of a job, started task, or TSO user to be monitored for data set events
ON JOBARRIV (JA)	Job name (or mask) of a job or started task that arrived on the JES spool from any source
ON JOBEND (JE)	Job name (or mask) of a job or started task that terminated
ON STEP (S)	Name (or mask) of a procedure step (and optionally, program step) to be monitored for termination

The And/Or/Not subparameter is always displayed in each specified ON statement. It is a conjunctional parameter for linking ON statements. Optional.

Entering a value for this subparameter opens a new ON statement and links the newly opened statement to the current ON statement.

When multiple ON statements are entered, the combinations of ON statements that can satisfy the selection criteria depend on the And/Or/Not values linking those ON statements. The logic applied to And/Or/Not subparameters is described in "General Information", which follows.

#### Valid values are:

- A (And) indicates AND logic between the two statements If both ON statements are true, the event criteria are satisfied.
- O (Or) indicates OR logic between the preceding and following ON statements If either statement is true, the event criteria are satisfied.
- N (Not) indicates AND NOT logic between the two statements If the prior statement is true and the subsequent statement is false, the event criteria are satisfied.

### **General Information**

Upon typing an ON parameter and pressing **Enter**, additional fields (subparameters) are displayed. Each ON parameter and its subparameters comprise an ON statement. At least one ON statement is required in a rule definition. Additional ON statements can be entered using the And/Or/Not option.

The first eight characters of the event name appear as the name of the rule in the Rule List screen.

### **And/Or/Not Subparameter Logic**

The following logic is applied:

- AND and NOT logic are applied before OR logic
- NOT means AND NOT as represented below
- A NOT B is interpreted as A AND (NOT B)
- A OR B AND C is interpreted as A OR (B AND C)
- A AND B OR C NOT D is interpreted as [(A AND B) OR (C AND NOT D)]

Use of OR logic reduces the amount of redundant data in the CMEM rule library and improves rule management.

### – NOTE –



When entering multiple ON statements, ensure that the statements are not mutually exclusive or not connected by an AND parameter. Rules containing mutually exclusive ON statements connected by an AND parameter are never triggered. For example:

ON DSNEVENT JOBA STEPA

AND

ON DSNEVENT JOBA STEPB

### **Character Masking**

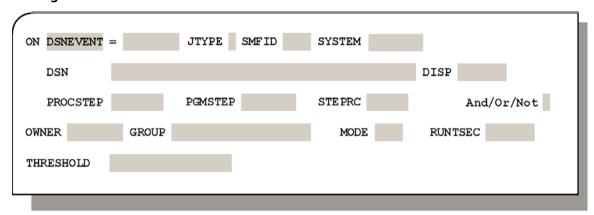
The following mask characters can be used when entering ON statement values:

- * represents any number of characters, including no characters
- ? represents any one character

## **ON DSNEVENT: Event Parameter**

Monitor a data set disposition event.

Figure 343 ON DSNEVENT Parameter Format



Optional. Type D (DSNEVENT) in the ON field and press **Enter**. The following subparameters are displayed:

Table 230 DSNEVENT Subparameters (part 1 of 3)

Subparameter	Description	
jobname	Name (or mask) of the job to be monitored for data set events. Mandatory.	
JTYPE	Type of job to be monitored for data set events:  ■ J (Job) – batch job ■ S (STC) – started task ■ T (TSU) – TSO user ■ blank – any type of job; valid only if STEPRC is blank, that is, the rule is processed immediately upon detection of the data set event. Default.  If JTYPE is entered, only a data set event occurring in a job of the specified type can trigger the rule.	
SMFID	SMF ID of the CPU to monitor for data set events. Mask characters (* and ?) are supported. Default: current CPU.	
SYSTEM	Name of the system to monitor for data set events. Mask characters (* and ?) are supported. Default: current system.	

Table 230 DSNEVENT Subparameters (part 2 of 3)

Subparameter	Description
DSN	Name of data set (or mask) to be monitored for this event within the selected jobs. Mandatory. Valid values are:
	■ DISP – data set disposition
	Mandatory. The abbreviation (that is, the first letter) of the desired value can be entered. Valid values are:
	<ul> <li>CATLG – cataloged (including SMS-managed files and ROLLED-IN SMS-managed GDG files)</li> <li>DELETE – deleted</li> </ul>
	<ul> <li>— UNCATLG – uncatalogued</li> <li>— KEEP – kept (including SMS-managed files)</li> <li>— RETAIN – cataloged or kept</li> </ul>
	<ul> <li>SCRATCH - deleted and uncatalogued (SMS managed files)</li> <li>ALL - any of the above dispositions</li> <li>NCT2 - occurrence of a NOT CATLG 2 event - when a data</li> </ul>
	set was created in a previous job step, but not cataloged at deallocation because its name already exists in the MVS catalog
	— * – any of the above data set dispositions (including NCT2)
PROCSTEP	Name or mask of a step invoking a procedure or, for a started task, task ID. Optional.
	If omitted, all procedure steps in the selected jobs are monitored.
	<b>Note:</b> When a started task is initiated, it can be assigned a task ID. For example, in command S GTF.G, the task ID of GTF is G. If a task ID is not entered, MVS assigns a default task ID to the started task, as follows:
	■ For a system started task with stepname IEFPROC, MVS sets an internal task ID.
	■ For other started tasks, the default task ID equals the procedure (started task) name.
	Therefore, when using CMEM to monitor system started tasks, if no task ID is entered in the START command, the PROCSTEP parameter must not be specified.
PGMSTEP	Name (or mask) of a step invoking a program. Optional. If omitted, all program steps in the selected jobs are monitored.
	Note: When a system started task with stepname IEFPROC is initiated, MVS assigns the step a default program step name. Therefore, when using CMEM to monitor these system started tasks, the PGMSTEP parameter must not be specified.

Table 230 DSNEVENT Subparameters (part 3 of 3)

Subparameter	Description
STEPRC	Determines at which point in the job step, and under what conditions in the job step, the DO statements are performed. Valid values are:
	<ul> <li>blank – if no completion code is entered, the rule is executed immediately upon detection of the specified data set event If any of the following values is entered for STEPRC, execution of the DO statements is delayed until the end of the monitored job step and is dependent upon how the jobstep ended:         <ul> <li>OK – step ended with a condition code of zero</li> <li>NOTOK – step ended with a nonzero code</li> <li>***** – step ended (with any code)</li> <li>Cnnnn – step ended with the indicated condition code</li> <li>Snnn – step ended with the indicated system abend code</li> <li>Unnnn – step ended with the indicated user abend code</li> </ul> </li> <li>Asterisks can be entered instead of code digits; condition codes and abends can be preceded by code qualifiers (&lt;, &gt;, N). For more information, see the following section, "General Information".</li> </ul>
And/Or/Not	Conjunctional parameter that opens a new ON statement and links it to the previous ON statement. Optional. Valid values are:  ■ A (And) – indicates AND logic between the two ON statements  ■ O (Or) – indicates OR logic between the preceding and following sets of ON statements  ■ N (Not) – indicates AND NOT logic between the two ON statements

### **General Information**

ON DSNEVENT rules are triggered by the setting of data set disposition at time of deallocation (during step termination or dynamic deallocation).

DO statements in the rule are executed either immediately upon detection of the data set event or at the end of the job step that caused the data set event, depending on the value entered in the STEPRC subparameter (described above).

Immediate execution is useful for performing actions when data sets are dynamically de-allocated using long running address spaces (for example, CICS, TSO users, and file transfer monitors).

CMEM must be active before any tasks tracked by ON DSNEVENT rules begin; moreover, ON DSNEVENT rules only intercept data set events for jobs, started tasks, or TSO users that started after the rule was ordered.

#### — NOTE



To monitor data set events for a job, started task or TSO user, the job, started task or TSO user must have MSGLEVEL=(1,1) and message IEF403I or IEF125I must appear in the job log.

A DSNEVENT can be triggered only on the CONTROL-O/CMEM that runs on the same z/OS system on which the event occurs. Even a focal CONTROL-O or CMEM in a Sysplex environment is insufficient. This is because CONTROL-O or CMEM intercepts the messages written to JESYSMSG while they are being written, which can be done only on the same system where the event happens.

ON DSNEVENT rules do not intercept data set events, such as cataloging, uncataloging, or scratching, when they are performed using MVS CATALOG or SCRATCH macros.

The following restrictions apply to ON DSNEVENT statements:

- Do not specify an ON DSNEVENT statement with any other type of ON statement in a rule.
- Do not specify different STEPRC values in the same rule. If you do, the last specified value is used.

Entering values for the optional subparameters PROCSTEP, PGMSTEP and STEPRC limits the situations that can satisfy the step termination event. Conversely, if a subparameter is blank, that subparameter is ignored.

### **Example**

- If a PGMSTEP and PROCSTEP value are both entered, the rule is triggered only if the specified PGMSTEP is completed in the specified PROCSTEP.
- If a PGMSTEP value is entered without a PROCSTEP value, the rule is triggered if the PGMSTEP is completed anywhere within the job stream.

### The STEPRC Subparameter

When entering a condition code or abend code in the STEPRC subparameter, any characters in the code can be replaced by an asterisk (*). An asterisk means "any value" for the character it replaces. For example, if S*13 is entered, the code criteria is satisfied by codes S013, S613, S913, and so on.

When entering condition and/or abend codes, the following qualifiers can be used as indicated:

**Table 231 Valid STEPRC Code Qualifiers** 

Qualifier	Description
<	Greater than. Valid for condition codes and user abend codes.
>	Less than. Valid for condition codes and user abend codes.
N	Triggers the rule if the specified code does not exist in the step. Valid as a qualifier for condition codes, user abend codes, and system abend codes.

### The SMFID and SYSTEM Subparameters

The default values for the SMFID and SYSTEM subparameters are the current system. If no value is entered for either SMFID or SYSTEM, the rule is triggered only by events that occur in the current system.

## **Example**

When a new production data set is created, trigger a backup job.

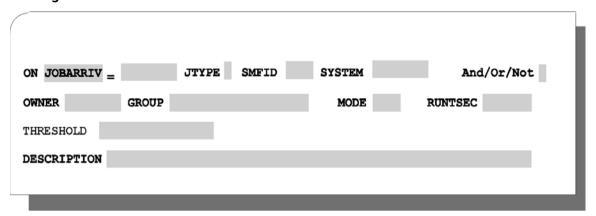
Figure 344 ON DSNEVENT Parameter Example

```
RL: PRDJ0003 LIB CTM.PROD.RULES
                                                          TABLE: BACKUP
COMMAND ===>
                                                         SCROLL===> CRSR
  ON DSNEVENT = PRDJ0003 JTYPE SMFID SYSTEM
  DSN PROD.* DISP CATLG
PROCSTEP PGMSTEP STEPRC OK And/C
OWNER ADMIN GROUP BACKUP MODE PROD RUNTSEC
                                                      And/Or/Not
  THRESHOLD
  DESCRIPTION NEW DATASET CREATED - TRIGGER A BACKUP JOB
  /* SCHEDULE A CONTROL-M JOB TO HANDLE THE BACKUP
  /*
  DO FORCEJOB = TABLE BACKUP JOB BACKUP DATE ODAT
              LIBRARY CTM.PROD.SCHEDULE
  D0
  ==== >>>>>>>> END OF RULE DEFINITION PARAMETERS <<<<<<<<<<<
FILL IN RULE DEFINITION. CMDS: CAPS, EDIT, SHPF,
```

## **ON JOBARRIV: Event Parameter**

Monitor a job arrival event on the JES spool.

Figure 345 ON JOBARRIV Parameter Format



Optional. Type JA (JOBARRIV) in the ON field and press **Enter**. The following subparameters are displayed:

**Table 232 ON JOBARRIV Subparameters** 

Subparameter	Description		
jobname	Job name (or mask). Mandatory. For more information, see "Character Masking" on page 83.		
JTYPE	<ul> <li>Type of job that can trigger the rule. Optional. Valid values are:</li> <li>J (JOB) – batch job</li> <li>S (STC) – started task</li> <li>T (TSU) – TSO user</li> <li>blank – if no value is entered, the rule can be triggered by any type of job. Default.</li> </ul>		
SMFID	SMF ID of the CPU to monitor for job arrival events. Mask characters (* and ?) are supported. Default: current CPU.		
SYSTEM	Name of the system to monitor for job arrival events. Mask characters (* an ?) are supported. Default: current system.		
And/Or/Not	Conjunctional parameter that opens a new ON statement and links it to the previous ON statement. Optional. Valid values are:  A (And) – indicates AND logic between the two ON statements O (Or) – indicates OR logic between the preceding and following sets of ON statements N (Not) – indicates AND NOT logic between the two ON statements		

## **General Information**

ON JOBARRIV statements can be used to trigger CONTROL-M actions based on the appearance of a job on the JES spool.

Combination of an ON JOBARRIV statement and a DO FORCEJOB statement can be used to control an external job through CONTROL-M. Such a job is called an On Spool job. For more information, see "On Spool Jobs" on page 683.

The default values for the SMFID and SYSTEM subparameters are the current system. If no value is entered for either SMFID or SYSTEM, the rule is triggered only by events that occur in the current system.

### — NOTE —



For JES3 users: JOBARRIV rules are processed on the global CPU.

## **Example**

Backup jobs submitted outside CONTROL-M must be monitored by CONTROL-M.

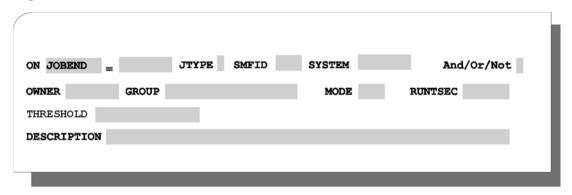
### Figure 346 ON JOBARRIV Parameter Example

```
RI: BKP*
             LIB CTM.PROD.RULES
                                                            TABLE: BACKUP
COMMAND ===>
                                                            SCROLL===> CRSR
  ON JOBARRIV = BKP* JTYPE SMFID SYSTEM
OWNER ADMIN GROUP BACKUP MODE PROD
                                                             And/Or/Not
                                           MODE PROD RUNTSEC
  DESCRIPTION MONITOR EXTERNAL BACKUP JOBS
  DESCRIPTION
  /* TELL CONTROL-M TO MONITOR THIS JOB
  DO FORCEJOB = TABLE BACKUP JOB
                                                           DATE ODAT
               LIBRARY CTM. PROD. SCHEDULE
  /*
 ===== >>>>>>>> END OF RULE DEFINITION PARAMETERS <<<<<<<<< =====
FILL IN RULE DEFINITION. CMDS: CAPS, EDIT, SHPF
```

## **ON JOBEND: Event Parameter**

Monitor a job termination event.

Figure 347 ON JOBEND Parameter Format



Optional. Type JE (JOBEND) in the ON field and press **Enter**. The following subparameters are displayed:

**Table 233 JOBEND Subparameters** 

Subparameters	Description
jobname	Job name (or mask). Mandatory.
JTYPE	Type of job whose termination can trigger the rule. Optional. Valid values are:
	J (JOB) – batch job
	■ S (STC) – started task
	<ul> <li>T (TSU) – TSO user</li> <li>blank – if no value is entered, the rule can be triggered by the termination of any type of job. Default.</li> </ul>
SMFID	SMF ID of the CPU to monitor for job termination events. Mask characters (* and ?) are supported. Default: current CPU.
SYSTEM	Name of the system to monitor for job termination events. Mask characters (* and ?) are supported. Default: current system.
And/Or/Not	Conjunctional parameter that opens a new ON statement and links it to the previous ON statement. Optional. Valid values are:
	<ul> <li>A (And) – indicates AND logic between the two ON statements</li> <li>O (Or) – indicates OR logic between the preceding and following sets of ON statements</li> </ul>
	<ul><li>N (Not) – indicates AND NOT logic between the two ON statements</li></ul>

### **General Information**

ON JOBEND statements can be used to trigger CONTROL-M actions based on the termination of a job.

The default values for the SMFID and SYSTEM subparameters are the current system. If no value is entered for either SMFID or SYSTEM, the rule is triggered only by events that occur in the current system.

### NOTE -



For JES3 users: JOBEND rules are processed on the same CPU that executed the specified job.

## **Example**

Instruct CONTROL-M to start the batch shift when CICSPROD ends.

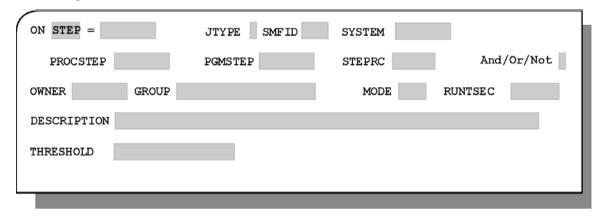
### Figure 348 ON JOBEND Parameter Example

RL: CICSPROD LIB CTM.PROD.RULES  COMMAND ===>	TABLE: STCS SCROLL===> CRSR
ON JOBEND = CICSPROD JTYPE SMFID SYSTEM OWNER ADMIN GROUP CICS MODE PROD DESCRIPTION START THE BATCH SHIFT DESCRIPTION	
/* INFORM CONTROL-M THAT THE BATCH CAN BE STARTED  DO COND = START-BATCH ODAT +  DO	
====== >>>>>>>> END OF RULE DEFINITION PARAMETERS	
FILL IN RULE DEFINITION. CMDS: CAPS, EDIT, SHPF	21.00.36

## **ON STEP: Event Parameter**

Monitor a job step termination event.

Figure 349 ON STEP Parameter Format



Optional. Type S (STEP) in the ON field and press **Enter**. The following subparameters are displayed:

Table 234 ON STEP Subparameters (part 1 of 2)

Subparameter	Name (or mask) of the job to be monitored for step termination.  Mandatory.		
job			
JTYPE	Type of job to be monitored for step termination. Optional. Valid values are:		
	■ J (Job) – batch job		
	■ S (STC) – started task		
	■ T (TSU) – TSO user task		
	■ blank – any type of job. Default.		
	If JTYPE is entered, only the termination of steps from the specified type of job can trigger the rule.		
SMFID	SMF ID of the CPU to monitor for data set events. Mask characters (* and ?) are supported. Default: current CPU.		
SYSTEM	Name of the system to monitor for data set events. Mask characters (* and ?) are supported. Default: current system.		
PROCSTEP	Name (or mask) of a step invoking a procedure or, for a started task, task ID. Optional. If omitted, all procedure steps in the selected jobs are monitored.		

Table 234 ON STEP Subparameters (part 2 of 2)

Subparameter	Description
	Note: When a started task is initiated, it can be assigned a task ID. For example, in command S GTF.G, the task ID of GTF is G. If a task ID is not entered, MVS assigns a default task ID to the started task, as follows:
	<ul> <li>For a system started task with stepname IEFPROC, MVS sets an internal task ID.</li> <li>For other started tasks, the default task ID equals the procedure (started task) name.</li> </ul>
	Therefore, when using CMEM to monitor system started tasks, if no task ID is entered in the START command, do not specify the PROCSTEP parameter.
PGMSTEP	Name (or mask) of a step invoking a program. Optional. If omitted, all program steps in the selected jobs are monitored.
	<b>Note:</b> When a system started task with stepname IEFPROC is initiated, MVS assigns the step a default program step name. Therefore, when using CMEM to monitor these system started tasks, do not specify the PGMSTEP parameter.
STEPRC	Return codes and/or statuses returned upon termination of the specified steps that satisfy the step termination criteria. Valid values are:
	<ul> <li>' '(Blank) or **** - step ended with any code or status         If no value or four asterisks are entered, the return code or status         is irrelevant.</li> <li>OK - step ended with a condition code of 0</li> <li>NOTOK - step ended with a nonzero code</li> <li>Cnnnn - step ended with the indicated condition code</li> <li>Snnn - step ended with the indicated system abend code</li> <li>Unnnn - step ended with the indicated user abend code</li> </ul>
	Asterisks can be entered instead of code digits; condition codes and abends can be preceded by code qualifiers (<, >, N). For more information, see the following section, "General Information".
And/Or/Not	Conjunctional parameter that opens a new ON statement and links it to the previous ON statement. Valid values are:
	<ul> <li>A (And) – indicates AND logic between the two ON statements</li> <li>O (Or) – indicates OR logic between the preceding and following sets of ON statements</li> <li>N (Not) – indicates AND NOT logic between the two ON statements</li> </ul>

### **General Information**

ON STEP rules are triggered when specified job steps terminate with specified return codes or statuses.

Entering values for the optional subparameters PROCSTEP, PGMSTEP and STEPRC limits the situations that can satisfy the step termination event. Conversely, if a subparameter is blank, that subparameter is ignored.

- If a PGMSTEP and PROCSTEP value are both entered, the rule is triggered only if the specified PGMSTEP is completed in the specified PROCSTEP.
- If a PGMSTEP value is entered without a PROCSTEP value, the rule is triggered if the PGMSTEP is completed anywhere within the job stream.

CMEM must be active before any tasks tracked by ON STEP rules begin; moreover, ON STEP rules only intercept data set events for jobs, started tasks, or TSO users that started after the rule was ordered.

#### NOTE -



To monitor data set events for a job, started task or TSO user, the job, started task or TSO user must have MSGLEVEL=(1,1) and message IEF403I or IEF125I must appear in the job log.

A DSNEVENT can be triggered only on the CONTROL-O/CMEM that runs on the same z/OS system on which the event occurs. Even a focal CONTROL-O or CMEM in a Sysplex environment is insufficient. This is because CONTROL-O or CMEM intercepts the messages written to JESYSMSG while they are being written, which can be done only on the same system where the event happens.

### The SMFID and SYSTEM Subparameters

The default values for the SMFID and SYSTEM subparameters are the current system. If no value is entered for either SMFID or SYSTEM, the rule is triggered only by events that occur in the current system.

### The STEPRC Subparameter

When entering a condition code or abend code in the STEPRC subparameter, any characters in the code can be replaced by an asterisk (*). An asterisk means "any value" for the character it replaces. For example, if S*13 is entered, the code criteria is satisfied by codes S013, S613, S913, and so on. When entering condition and/or abend codes, the following qualifiers can be used as indicated:

**Table 235 ON STEP Subparameter STEPRC Qualifiers** 

Qualifier	Description
>	Greater than. Valid for condition codes and user abend codes.
<	Less than. Valid for condition codes and user abend codes.
N	Triggers the rule if the specified code does not exist in the step. Valid as a qualifier for condition codes, user abend codes, and system abend codes.

## **Example**

When step STEP2 in job PRD00010 is completed, add a prerequisite condition indicating that a file has been created.

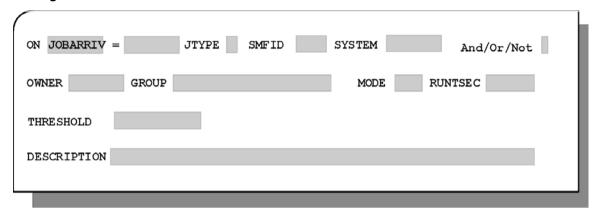
Figure 350 ON STEP Parameter Example

RL: PRD00010 LIB CTM.PRC			TABLE: CMEMRULE SCROLL===> CRSR
ON STEP = PRD00010 C PROCSTEP STEP2 F OWNER CTMCTLM GROUP THRESHOLD DESCRIPTION FOLLOWING ST DESCRIPTION INDICATING T DESCRIPTION	OTYPE SMFID PGMSTEP  TEP2 IN JOB PRDOOG	SYSTEM STEPRC OK MODE PROD	RUNTSEC NONE
DO COND = FILE-CREAT	ED ODAT -		
====== >>>> END	OF RULE DEFINIT:	ION PARAMETERS <	······································
FILL IN RULE DEFINITION. (	CMDS: CAPS, EDIT,	SHPF	21.00.36

## **OWNER: General Parameter**

Identifies the user requesting CMEM services.

Figure 351 OWNER Parameter Format



Mandatory. The OWNER parameter must be 1 through 8 characters.

### **General Information**

The OWNER parameter is primarily used by the internal security mechanism of CMEM to determine, together with an external security product, such as TOP SECRET, RACF or ACF2, those operations each user is authorized to perform.

## **Example**

The INCONTROL administrator is authorized to use the rule used to monitor the arrival of backup jobs.

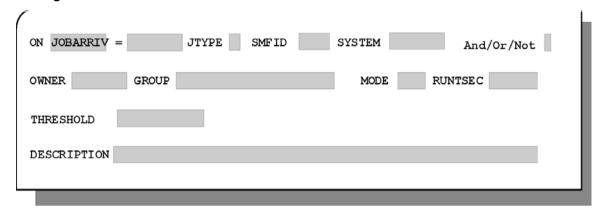
Figure 352 OWNER Parameter Example

```
RL: BKP* LIB CTM.PROD.RULES
                                              TABLE: BACKUP
COMMAND ===>
                                             SCROLL===> CRSR
+------
 ON JOBARRIV = BKP* JTYPE SMFID SYSTEM And/Or/Not
 OWNER ADMIN GROUP BACKUP
                               MODE PROD RUNTSEC
 THRESHOLD
 DESCRIPTION MONITOR STARTUP OF BACKUP JOBS
 DESCRIPTION
 /* TELL CONTROL-M TO MONITOR THIS JOB
 DO FORCEJOB = TABLE BACKUP JOB
                                      DATE ODAT
           LIBRARY CTM.PROD.SCHEDULE
 /*
 D0
 ----- >>>>>>> END OF RULE DEFINITION PARAMETERS <<<<<<<<< ----
FILL IN RULE DEFINITION. CMDS: CAPS, EDIT, SHPF
                                                   21.00.36
```

## **RUNTSEC:** General Parameter

Specifies the runtime security environment for the rule.

Figure 353 RUNTSEC Parameter Format



Optional. The abbreviation, that is, the first letter, of the desired value can be entered. Valid values are:

**Table 236 Valid RUNTSEC Values** 

Value	Description
NONE	Runtime security checks are not performed for this rule.
OWNER	Runtime security checks are performed using the user ID entered in the OWNER field of the rule.
TRIGGER	Runtime security checks are performed using the user ID associated with the started task, TSO user, or batch job that invoked the rule.
''(Blank)	If CONTROL-O is not active, the default is OWNER.  If CONTROL-O is active, performance of runtime security checks depend on the value of the Global parameter RUNTDFT (NONE, OWNER, or TRIGGER) in the CTOPARM member as entered at time of CONTROL-O installation.

### NOTE



Value TRIGGER applies only to ON DSNEVENT, ON STEP, or ON JOBEND event rules. If the value TRIGGER is entered for an ON JOBARRIV event rule, the value is treated as OWNER.

## **General Information**

The RUNTSEC parameter is used by the CMEM security interface for interaction with external security products, such as CA-RACF, CA-TOP SECRET, and CA-ACF2. For more information see the *INCONTROL for z/OS Security Guide*.

CMEM security checks are carried out in two stages: at order time and at runtime.

At order time, security checks are carried out to ascertain whether the owner of the rule, as specified in the OWNER subparameter, is authorized to code each one of the rule statements.

At runtime, additional security checks are carried out to determine whether the user who owns the rule (RUNTSEC=OWNER) or the user who triggered the rule (RUNTSEC=TRIGGER) is authorized to execute a DO COND or DO FORCEJOB statement defined in the rule.

### **Example**

Perform a backup using the security ID of the job that triggered the rule.

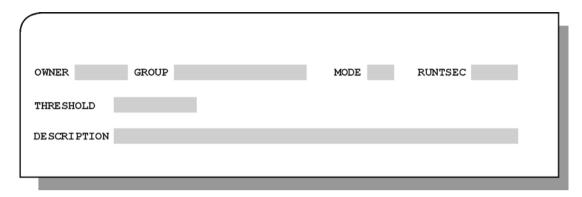
Figure 354 RUNTSEC Parameter Example

```
RL: PRDJ0003 LIB CTM.PROD.RULES
                                                            TABLE: BACKUP
COMMAND ===>
                                                            SCROLL===> CRSR
  ON DSNEVENT = PRDJ0003 JTYPE SMFID SYSTEM
  DSN PROD.* DISP CATLG
PROCSTEP PGMSTEP STEPRC OK And/Or/Not
OWNER ADMIN GROUP BACKUP MODE PROD RUNTSEC TRIGGER
  THRESHOLD
  DESCRIPTION NEW DATASET CREATED - TRIGGER A BACKUP JOB
  DESCRIPTION
  /* SCHEDULE A CONTROL-M JOB TO HANDLE THE BACKUP
  DO FORCEJOB = TABLE BACKUP JOB BACKUP
                                              DATE ODAT
               LIBRARY CTM.PROD.SCHEDULE
  D0
 ----- >>>>>>>> END OF RULE DEFINITION PARAMETERS <<<<<<<< ----
FILL IN RULE DEFINITION. CMDS: CAPS, EDIT, SHPF,
                                                                   21.00.36
```

## **THRESHOLD: Runtime Scheduling Parameter**

Limits the number of times that a rule can be triggered in one CMEM Monitor cycle.

Figure 355 THRESHOLD Parameter Format



Optional. Valid values are 1 through 999999999.

### **General Information**

The THRESHOLD parameter is used to prevent unlimited loops within the system. The value assigned to the parameter indicates the maximum number of times that CMEM will trigger the rule in a single CMEM interval.

Before CMEM triggers a rule, it determines whether the rule has already been triggered the maximum number of times in the current CMEM interval. If so, CMEM does not trigger the rule again, but instead sets the STATUS of the rule to SUSPEND and issues message CTO285W to the console.

If no value, or a value of 0, is entered for THRESHOLD, CMEM does not limit the number of times that the rule can be triggered.

## **Example**

Limit execution of the following rule to 10 executions. Do not allow the rule to be triggered until it is released from SUSPEND status.

Figure 356 THRESHOLD Parameter Example

RL: PRDJOBO1 LIB CTM.PROD.RULES  COMMAND ===>	TABLE: JOB SCROLL===> CRSR
ON JOBARRIV = JOBNM233 JTYPE SMFID SYSTEM	And/Or/Not RUNTSEC
DO COND = JOBNX-ARRIVED ODAT +	
====== >>>>>>>>>>>> END OF RULE DEFINITION PARAMETER	S <<<<<< ====
=====FILL IN RULE DEFINITION. CMDS: CAPS, EDIT, SHPF	13.21.53





# **JCL and AutoEdit Facility**

This chapter includes the following topics:
Overview
System variables
Non-Date System Variables
Date System Variables
Special System Variables
User-Defined Variables
Valid Characters in User-Defined Variables
Local Variables757
Global Variables
JCL Setup Operation Flow
Rules of Variable Resolution
Order of Precedence for Multiple Value Assignments
Control Statements
%%GLOBAL771
%%GOTO and %%LABEL772
%%IF, %%ELSE, %%ENDIF
%%INCLIB and %%INCMEM
%%LIBSYM and %%MEMSYM776
%%RANGE777
%%RESOLVE778
%%SET
Operators
Functions
%%\$CALCDTE
%%\$GREG
%%\$JULIAN
%%\$LEAP
%%\$WCALC
%%\$WEEK#
%%\$WEEKDAY
%%\$YEARWK#787
%%CALCDATE
%%SUBSTR
0/0/6LENICTLI

%%\$TYPE	700
%%\$FUNC	
Testing AutoEdit Syntax	
AutoEdit Usage in the Job Scheduling Definition	795
Examples	797
Date Variables	797
ODATE, RDATE and DATE Usage	797
How to Obtain Date Formats – 1	
How to Obtain Date Formats – 2	
How to Obtain Date Formats – 3	800
How to Obtain the Previous Month's Last Business Date	800
Automatic Job Order for the Next Day	801
Tape Clearance System – Stage 1	802
Tape Clearance System – Stage 2	
Tape Management System	
Dynamic Job Name	
Controlling the Target Computer by Class	
Controlling the Target Computer by System Affinity	
%%BLANKn Statement	
%%RANGE Statement	
SYSIN Parameter Containing %%	
%%INCLIB and %%INCMEM Statements	
Rooloan "IF" Logic	808

## **Overview**

In the production environment, it is often necessary to manually modify JCL prior to submission of a job, as in the following cases:

- changing a parameter or a date card
- supplying tape numbers in JCL procedures
- eliminating steps under different run conditions (for example, end of month processing versus normal daily run)

Manual modification of JCL is inconvenient at best, and it may, in fact, be error prone and lead to serious problems.

The AutoEdit facility offers an automated alternative to manual JCL modification. This facility permits AutoEdit terms (variables, functions, and similar terms, described in this chapter) to be specified in the JCL in place of values that change from job submission to job submission. At time of job submission, these terms are resolved to their actual values.

The inclusion of AutoEdit terms in the job stream can eliminate the need to continually change the JCL. Some AutoEdit terms can also be used in job scheduling definitions.

AutoEdit terms are prefaced by a %% symbol, which distinguishes them from terms that are not AutoEdit terms. For example, the term %%ODAY is recognized as an AutoEdit term.

#### NOTE



AutoEdit terms must be placed within the job stream submitted by CONTROL-M, not within a catalogued JCL procedure.

The use of AutoEdit terms within started tasks (STCs) is not supported.

The components of the AutoEdit facility are described briefly on the following pages and in greater detail later in this chapter.

### **Variables**

Variables are the main components of the AutoEdit facility. Variables are used to replace manually changed values, generally within the JCL. AutoEdit variables can be either of the following types:

System Variables

System variables are predefined, reserved variables that represent information about the system.

For example, System variable %%ODATE is replaced by the job's original scheduling date.

#### User-Defined Variables

User-defined variables are created by the user. The user must provide the value (or the tools to derive the value) that replaces the variable at time of job submission.

For example, the user can define a variable, %%SPACE-TYPE, to represent the type of storage unit (cylinder or track) on disk.

User-defined variables are either:

#### local variables

Local variables are used only within the job stream. The value of a local variable can be set or changed within the job stream by CONTROL-M, but the changed value is kept only in memory for use during submission of that job stream. The value is not passed to another job stream.

### Global variables

Global variables are user-defined variables that are placed in the IOA Global Variable database, from which they can be accessed and updated by other CONTROL-M jobs and CONTROL-O rules.

System variables and user-defined variables are discussed in detail below. Local variables and Global variables are also discussed in detail, under the topic of user-Defined variables.

### control Statements

AutoEdit control statements in the JCL define the environment for user-defined variables. The AutoEdit facility supports many AutoEdit control statements, and this is discussed in detail later. Some of the more important Control Statements are described here briefly.

**Table 237 AutoEdit Control Statements** 

Statement	Description
%%GLOBAL member	Identifies a member that contains a set of user-defined local variables and their assigned values.
%%LIBSYM library / %%MEMSYM member	Identifies a member and library that contain a set of user-defined local variables and their assigned values.
%% SET %%variable = value	Sets a value to a user-defined variable in the JCL.

### **Operators**

AutoEdit operators modify the values of AutoEdit variables in the JCL. For example, in the following statement, operator %%PLUS assigns a number to a scratch tape that is one higher than the previously assigned tape number:

```
//* %%SET %%SCRATCH=%%SCRATCH %%PLUS 1
```

#### **Functions**

AutoEdit functions perform functions on specified AutoEdit variables in the JCL. For example, in the following statement, the %%CALCDATE function sets AutoEdit variable %%NEXTDAY to one day after the current System variable %%ODATE:

```
//* %%SET %%NEXTDAY=%%CALCDATE %%ODATE +1
```

The format of a function statement depends on the function.

### **AutoEdit Components in the Job Scheduling Definition**

Although the most important and common use of AutoEdit components is in JCL setup, certain AutoEdit components can also be used in the job scheduling definition.

SET VAR and DO SET job scheduling definition statements assign values to user-defined variables. These statements perform a similar function as, and work together with, %%SET control statements specified in the JCL.

Other job scheduling definition statements (for example, SYSOUT, SHOUT, MEMLIB) allow specification of System variables.

### **JCL Setup and AutoEdit Features**

The rest of this chapter contains a description of the following JCL Setup and AutoEdit topics:

- System variables
- **■** User-Defined Variables
- JCL Setup Operation Flow
- Rules of Variable Resolution
- Control Statements
- Operators
- Functions
- Testing AutoEdit Syntax
- AutoEdit Usage in the Job Scheduling Definition
- Examples

## **System variables**

CONTROL-M System variables are predefined, reserved, commonly used variables whose values are automatically updated and maintained by CONTROL-M. The System variable format is:

%%var

where var is the name of the System variable.

Each variable resolves to the corresponding system value. The length of the line is changed accordingly. For example:

```
//EJ%%ODATE JOB (0,15, ...
// EXEC ACCOUNTS,DAY=%%ODAY,MONTH=%%OMONTH
```

If the original scheduling date is June 5, 2001, the variables are resolved as follows:

```
//EJ000603 JOB (0,15, ...
// EXEC ACCOUNTS,DAY=05,MONTH=06
```

## **Non-Date System Variables**

The following AutoEdit System variables are supported by CONTROL-M (in both JCL and in job scheduling definitions):

Table 238 Non-Date AutoEdit System Variables (part 1 of 3)

Variable	Description	
%%.	Concatenation symbol.	
%%APPL	Application to which the job belongs.	
%%BLANK	Blank.	

Table 238 Non-Date AutoEdit System Variables (part 2 of 3)

Variable	Description
%%BLANKn	Resolves to <i>n</i> blanks, where <i>n</i> is a number from 1 through 80.
	<b>Note</b> : When entering an AutoEdit variable assignment (such as %%A = 1), any leading spaces (blanks) following the '=' are ignored and any succeeding spaces in the character string terminate the variable value when the variable is resolved. System variables %%BLANK and %%BLANK <i>n</i> enable you to embed spaces in the variable expression.
	The variable assignment %%A = This%%BLANKis%%BLANKlan%%BLANKexample, using %%BLANK and %%BLANKn to create embedded spaces, resolves to This is an example %%BLANK and %%BLANK1 each produce the same result, a single embedded space.
	The similar variable assignment, %%A = THIS IS AN EXAMPLE, resolves to THISISANEXAMPLE.
%%GROUP	Group to which the job belongs.
%%JOBNAME	Name of the submitted job as specified in the JCL job statement. If %%JOBNAME is resolved before the job submission (for example, %%JOBNAME is used in a SHOUT WHEN LATESUB statement, and the job has not been submitted), its value is assigned the %%\$MEMNAME value.
%%ORDERID	Unique job order ID under CONTROL-M (5 characters).
%%OWNER	Owner of the job, as specified in the scheduling definition.
%%RN	Run number (can exceed one for cyclic and rerun and restarted jobs).
%%TIME	Current time of day, in hhmmss format. This variable receives the current time each time it is invoked.
%%TIMEID	Current time of day, in hhmmss format. This AutoEdit variable receives the current time only when it is first resolved in a job. Subsequent resolutions within that job do not update the variable.
%%\$DATEFMT	Resolves to A, W, or J, according to the DATETYP date format parameter defined in the IOAPARM member in the IOA PARM library.
%%\$L	The computer LPAR name where the job executed.
%%\$MEMNAME	Name of the JCL member from which the job is submitted. (This corresponds to the value specified in the job scheduling definition.)
%%\$QNAME	Qname (unique identifier) of the monitor that submitted the job.
%%\$SCHDLIB	Name of the scheduling library that contains the job scheduling definition of the job. This variable is resolved only after the job has been ordered.
%%\$SCHDTAB	Name of the scheduling table that contains the job scheduling definition of the job. This variable is resolved only after the job has been ordered.

Table 238 Non-Date AutoEdit System Variables (part 3 of 3)

Variable	Description
%%\$SIGN	1-character ID of the computer on which the job is running. The %%\$SIGN variable is commonly used when assigning system affinity, as in the following example:  /*JOBPARM SYSAFF=CPU%%\$SIGN For more information, see "Controlling the Target Computer by System Affinity" on page 805.
	Note: The %%\$SIGN variable is not resolved unless the job scheduling definition from which the job is ordered contains a resource name with a \$ mask character used as a generic indicator. For more information, see "RESOURCE: Runtime Scheduling Parameter" on page 596.
%%\$SYSNAME	Name of the system on which the CONTROL-M monitor is running.
%%\$TAG	Name of the schedule tag by which the job was scheduled. If the Group scheduling table was forced, or if the job was scheduled based on basic scheduling criteria other than a schedule tag, this value resolves to blanks.

## **Date System Variables**

Many System variables specify dates, or parts of dates, in various formats. Therefore, it is useful to understand the categories of dates with which CONTROL-M is concerned.

Dates are divided into the categories listed below. For a description of these categories, see "Date Definition Concepts" on page 63

### ■ Working Date

System variables that specify working dates begin %%R (%%RDATE, %%RDAY, and so on).

### Original Scheduling Date

System variables that specify original scheduling dates begin %%O (%%ODATE, %%ODAY, and so on).

### ■ System Date

System variables that specify system dates have no special prefix other than %% (%%DATE, %%DAY, and so on).

Although these types of dates are resolved in Gregorian format, Julian formats can also be requested (%%JULDAY, %%OJULDAY and %%RJULDAY).

The following date AutoEdit System variables are supported by CONTROL-M in JCL | and in certain job scheduling definition parameters (for more information, see "AutoEdit Usage in the Job Scheduling Definition" on page 795):

**Table 239 Date AutoEdit System Variables** 

Variable	Description
%%\$CENT	First two digits in the current year (for example, 20 in the year 2000).
%%DATE	Current system date (format yymmdd).
%%DAY	Current system day (format dd).
%%MONTH	Current system month (format mm).
%%YEAR	Current system year (format yy).
%%WEEK	Current week in the year (that is, 01 through 53).
%%WDAY	Current system day of the week (Example: 1=Sunday, 2=Monday and 0=Saturday). ^a
%%\$OCENT	First two digits of the year in which the job was originally scheduled.
%%ODATE	Original scheduling date of the job (format yymmdd).
%%ODAY	Original scheduling day of the job (format dd).
%%OMONTH	Original scheduling month of the job (format mm).
%%OYEAR	Original scheduling year of the job (format yy).
%%OWEEK	Original scheduling week of the job (that is, 01 through 53).
%%OWDAY	Original scheduling day of the week of the job (format d; Example: 1=Sunday, 2=Monday and 0=Saturday).
%%\$RCENT	First two digits of the current working year.
%%RDATE	Current working date (format yymmdd).
%%RDAY	Current working day (format dd).
%%RMONTH	Current working month (format mm).
%%RYEAR	Current working year (format yy).
%%RWEEK	Current working week (that is, 01 through 53).
%%RWDAY	Current working day of the week (format d; Example: 1=Sunday, 2=Monday and 0=Saturday). 1
%%JULDAY	Current system day (Julian format jjj).
%%OJULDAY	Original scheduling day of the job in the year (Julian format jjj).
%%RJULDAY	Current working day of the year (Julian format jjj).

^a Start of the week at a site depends upon an IOA installation parameter that specifies whether 1=Sunday or 1=Monday. Your INCONTROL administrator can tell you whether the week begins on Sunday or Monday at your site. The above reference assumes 1=Sunday, 2=Monday, ... 6=Friday, 0=Saturday.

The following AutoEdit System variables, prefixed %%\$, resolve to dates having 4-character years:

**Table 240 4 Character Year Date AutoEdit System Variables** 

Variable	Description
%%\$DATE	Current system date (format yyyymmdd).
%%\$YEAR	Current system year (format yyyy).
%%\$ODATE	Original scheduling date of the job (format yyyymmdd).
%%\$OYEAR	Original scheduling year of the job (format yyyy).
%%\$RDATE	Current working date (format yyyymmdd).
%%\$RYEAR	Current working year (format yyyy).
%%\$JULDAY	Current system day (Julian format yyyyjjj).
%%\$OJULDAY	Original scheduling day of the job in the year (Julian format yyyyjjj).
%%\$RJULDAY	Current working day of the year (Julian format yyyyjjj).

## **Special System Variables**

Special System variables are resolved during specific parts of the life cycle of jobs.

The following are the types of special System variables:

- those that are resolved after a group is ordered but before any of the jobs in the group are ordered
- those that can only be resolved after the job has ended
- those that can only be resolved after job submission

Special System variables of the latter types, that is, those that can only be resolved after the job has ended and those that can only be resolved after job submission, can only be used with the post-processing parameters in the job definition, such as SHOUT and DO IF RERUN.

# Special System variables resolved after a group is ordered, but before job submission

The special System variable shown in Table 241 can only be resolved after a group is ordered, but before job submission.

Table 241 Special AutoEdit System variable resolved after a group is ordered but before job submission

%%\$GRID	■ Group Entity Order ID, primarily used within a job scheduling definition. At job ordering time, this value is resolved to the Order ID of the Group to which the job belongs.
	■ If the job does not belong to a Group, or if the job is forced from a Group table, %%\$GRID is resolved to "??????".
	<ul> <li>%%\$GRID is resolved in condition names, in IN, OUT or DO COND parameters.</li> </ul>
	%%\$GRID is especially useful in condition names, to enable unique identification of condition names that are added by multiple ordering of jobs from the same group table.

## Special System variables resolved after end of job

The special System variables in Table 242 can only be resolved after the job has ended. These variables contain a blank value if the job ends OK or if no step in the job was run.

Table 242 Special AutoEdit System Variables Resolved after Job End

Variable	Description
%%JOBCC	Job completion code that caused the job to end NOTOK.
%%MAXRC	Highest return code in the job execution. For abended jobs, this variable resolves to blanks.
%%\$NODEID	The value of the node name that appears in the Active Environment Zoom screen. This variable is only resolved for NJE jobs; for non-NJE jobs, it resolves to a null value.
%%STEP	Job program step and procedure step (if it is defined) that triggered the postprocessing instruction.
	Format: 8-character program step (including blanks if necessary), followed by the procedure step name (up to eight characters).
%%\$PGMSTEP	Job program step (equal to the first part of the %%STEP variable).
%%\$PRCSTEP	Procedure step (equal to the second part of the %%STEP variable).



#### NOTE -

The effect of an ON PGMST...DO OK statement, or of the value of the MAXCCOK parameter may be that CONTROL-M treats a non-zero return code of a step in a job as OK. In such a case, the non-zero return code of that step is ignored by CONTROL-M in calculating values for the %%JOBCC and %%MAXRC AutoEdit System variables.

## Special System variables resolved after job submission

The special System variable shown in Table 243 can only be resolved after job submission. This variable contains a blank value if the job ends OK or if no step in the job was run.

Table 243 Special AutoEdit System Variable Resolved after Job Submission

Variable	Description
%%JOBID	JES job number

## **User-Defined Variables**

The ability to specify user-defined variables provides additional flexibility. You can define your own variables and assign values to them. CONTROL-M automatically edits the job stream accordingly. This facility is especially useful when it is necessary to share parameters or other information (for example, tape numbers) among jobs.

CONTROL-M assumes that strings beginning with %% are user-defined variables, except those beginning with %%\$, which are reserved System variables.

For a list of all System variables, see "System variables" on page 748.

User-defined variables are defined as either:

- Local variables, which are discussed in "Local Variables" on page 757
- Global variables, which are discussed in "Global Variables" on page 760

Multiple AutoEdit variables can be joined with each other and with constants, and periods (.) are often part of this process (for example,

JOB_%%JOBID%%._ENDED_OK). This is discussed in more detail in "Rules of Variable Resolution" on page 767.

Backslashes (\) are used only in Global variable assignments, and differentiate Global variables from local variables. For more information, see "Global Variable Assignment and Syntax" on page 761.

Unlike System variables, which are predefined and which receive their values from the system at time of job submission, two steps are performed for utilizing user-defined variables:

- The first step consists of specifying (defining) user-defined variables, usually within the JCL, instead of values that require manual modification.
- The second step consists of providing values to replace the user-defined variables at time of job submission. (Since the values are not provided by the system, the user must specify the appropriate values.) It is permissible, however, for user-defined variables to take their values from System variables (for example, %%SET %%VERSION = %%ODATE).)

User defined AutoEdit variables in post-processing statements of the scheduling definition (DO MAIL, DO SHOUT, and so on) are resolved according to the following search order:

- SET VAR statements in the job's definition
- SET VAR statements in the group entity (for jobs belonging to a group)
- AutoEdit definitions loaded into the CONTROL-M AutoEdit cache

Note that during post-processing, it is impossible to refer to an external library member containing AutoEdit variable definitions (that is, through %%LIBSYM xxxxx, %%MEMSYM yyyy, or %%GLOBAL zzzz statements). The only external method of resolving user AutoEdit variables for post-processing is to load the member(s) into the AutoEdit cache (see "Loading %%GLOBAL Members to Cache" on page 758.

### **Valid Characters in User-Defined Variables**

When defining AutoEdit variables, only certain characters can be used. The validity of characters depends on the purpose for which they are being used.

### **AutoEdit Variable Names**

AutoEdit variable names have a maximum of 163 characters. The following characters can be used in AutoEdit variable names:

- any character from A through Z, both uppercase and lowercase
- any digits from 0 through 9
- the following special characters:
  - & (Ampersand)
  - \$ (Dollar)
  - _ (Underscore)
  - -# (Octothorp)
  - -@ (At)
- the following hexadecimal values:
  - from x'41' through x'49'
  - from x'51' through x'59'
  - from x'62' through x'69'
  - -x'71'



### — NOTE -

BMC Software recommends that you do not use non-display hexadecimal values.

As variable names beginning with \$ are reserved for BMC use, do not use \$ as the first character of user-defined variables.

### **AutoEdit Variable Value Fields**

Any characters can be used in AutoEdit variable value fields except the following:

- ' ' (Blank)
- the following hexadecimal values:
  - -x'00'
  - -x'FE'
  - x'FF'





BMC Software recommends that you do not use non-display hexadecimal values.

### **AutoEdit Variables in JCL**

In any JCL line that contains AutoEdit variables, do not use the following hexadecimal values unless their processing is excluded by a %%RANGE or a %%RESOLVE OFF statement:

- -x'00'
- x'FE'
- __ x'FF'

### **Global AutoEdit Variables**

The following characters have special meanings in Global AutoEdit variables:

- . (Period)
- \ (Backslash)

Do not use these characters in the variable name field unless you require the special meaning assigned to them.

### **Local Variables**

Local variables are user-defined variables that are only within the job stream. The value of a local variable can be changed within the job stream, but the changed value is kept only in memory for use during submission of that job stream. The value is not passed to another job stream.

Local variables can be defined in either of two ways:

■ by means of %%SET statements in the JCL and/or SET VAR and DO SET statements in the job scheduling definition

%%SET statements are described under "Control Statements" on page 770. SET VAR statements are described in "SET VAR: General Job Parameter" on page 621, and DO SET statements are described in "DO SET: Post-Processing Parameter" on page 465.

by placing the variables and their values in special variable members

Variable members are members dedicated to holding user-defined AutoEdit variables and their values. These variables and values in these members can be used by any number of CONTROL-M jobs or CONTROL-O rules that are given access. However, these jobs and rules cannot update these members.

Members containing user-defined variables can be identified in either of two ways:

— by a %%MEMSYM control statement

This member must reside in the library specified in the %%LIBSYM statement that must accompany the %%MEMSYM statement. (The control statements %%LIBSYM and %%MEMSYM are described "Control Statements" on page 770.) Any number of such variable members can be defined.

— by a %%GLOBAL control statement

This statement differs from the %%MEMSYM statement in that it does not have an accompanying %%LIBSYM statement. Instead, the library in which the %%GLOBAL member resides is pointed to by a DAGLOBAL DD statement.

For example, the user may specify variable %%BRANCH_TAPE in a JCL statement:

```
//SOO1.INPUT DD VOL=SER=%%BRANCH_TAPE
```

and the %%MEMSYM member (or %%GLOBAL member) that assigns values might contain the following variable definition:

```
%%BRANCH TAPE=045673
```

%%MEMSYM, %%LIBSYM and %%GLOBAL control statements are described in "Control Statements" on page 770.

## **Loading %%GLOBAL Members to Cache**

%%Global members can be placed in cache memory, from where they can be accessed as needed. If the members are placed in cache, the JCL accesses the contents from the cache, instead of accessing the members themselves.

This can be very advantageous if many jobs access %%Global members, because each access of the member increases I/O and processing overhead.

Only those %%GLOBAL members that are specifically requested are loaded to cache. Requests are generally made by listing the desired %%GLOBAL members in a special cache list member in the DAGLOBAL library. This cache list member (default name: CACHLST) is pointed to by the AECACHL parameter in the CTMPARM member in the IOA PARM library.

Members are listed in the cache list member in the following format:

```
%%GLOBAL memname
```

where memname is the name of the %%GLOBAL member in the Global library.

The cache list member can optionally contain the following control statement as its first non-comment statement:

%%RESOLVE ALLCACHE

This control statement affects AutoEdit processing only if an AutoEdit variable has not been resolved by searching the %%GLOBAL members identified in the job. The statement instructs CONTROL-M to continue the variable resolution process by checking all members loaded into cache. Members in cache are searched in the same sequence they are listed in the cache list member.

%%GLOBAL members are loaded to cache at time of CONTROL-M startup.

To reload %%GLOBAL members to cache between CONTROL-M startups or to stop using AutoEdit cache, see "Loading %%GLOBAL Members to Cache" on page 758, and the corresponding topic in the *INCONTROL for z/OS Administrator Guide*.

### Format of Variable Members

A variable member (referenced by %%GLOBAL or %%MEMSYM statements) must be a member in a partitioned data set with a record length of 80. It can contain the following types of lines:

- Remark line: Line starting with an asterisk (*) in column 1. Remark lines are not processed.
- Assignment line: Line that assigns a value to a variable. The format is:

%%varname=value

Any number of user-defined variables (and their values) can be defined in a variable member. To designate a null value, omit the value.

### **Example**

```
* Last banking day in each month

*

%%LAST_BANKING_DAY_0001=010131

%%LAST_BANKING_DAY_0002=010228

%%LAST_BANKING_DAY_0003=010330

%%LAST_BANKING_DAY_0004=010430

%%LAST_BANKING_DAY_0005=010531

%%LAST_BANKING_DAY_0006=010629

%%LAST_BANKING_DAY_0007=010731

%%LAST_BANKING_DAY_0008=010831

%%LAST_BANKING_DAY_0009=010928

%%LAST_BANKING_DAY_0010=011031

%%LAST_BANKING_DAY_0011=011130

%%LAST_BANKING_DAY_0012=011231
```

## **Global Variables**

A Global variable is a user-defined variable that is placed in the IOA Global Variable database.

%%SET statements in the JCL, and SET VAR or DO SET statements in the job scheduling definition, enable CONTROL-M jobs and Group entities to define Global variables and place them in the IOA Global Variable database.

However, since %%SET, SET VAR and DO SET statements also define local variables, a distinguishing factor is needed to differentiate Local Variables from Global variables. The distinguishing factor is provided by syntax, which is described in "Global Variable Assignment and Syntax" on page 761.

A Global variable from the IOA Global Variable database can be specified anywhere a local variable can be specified in the JCL or the job scheduling definition (SHOUT, DO SHOUT, SYSOUT, DO SYSOUT, MEMLIB and OVERLIB statements).

## **Structure of the IOA Global Variable Database**

The IOA Global Variable database has a hierarchical structure consisting of several levels. This structure mirrors the hierarchical structure of the CONTROL-M components of which a CONTROL-M job is a part.

The levels in the IOA Global Variable database structure, starting from the lowest, are as follows:

Table 244 IOA Global Variable Database Structure Levels

Level	Description
Variable	Global variable in the IOA Global Variable database.
Job	Name of the job (JCL member) that appears in the MEMNAME field of the job scheduling definition.
Group	Group to which the job belongs. The name of the group appears in the GROUP field of the job scheduling definition.
Application	Application to which the group and job and belong. The name of the application appears in the APPL field of the job scheduling definition.
Product	M (CONTROL-M).

The importance of this structure is discussed in the topic "Global Variable Assignment and Syntax" immediately below.

## **Global Variable Assignment and Syntax**

Whenever a job (or Group entity) creates a Global variable and places it in the IOA Global Variable database, it assigns an owner to the variable.

The job that creates the variable can make itself the owner (for example, JOBA defines a Global variable that is assigned to JOBA), but it does not have to do this. It can, instead, assign a different owner to the variable (for example, JOBA defines a Global variable that it assigns to GROUP_ABC).

In fact, when a Global variable is created, it can be assigned to any component (job, group, application, or even to CONTROL-M) in the database. It is this ability to assign variables that makes the structure of the IOA Global Variable database so important.

The hierarchical structure of the IOA Global Variable Database, described above, is similar to the directory and subdirectory structure in Unix and DOS. Therefore, the same path structure and syntax that is used to describe directories and subdirectories is used to define and identify Global variables.

Note the following points about Global variable assignment and syntax:

 Global variables are identified (and distinguished from Local variables) by a backslash.

#### **Example**

- Variable %%PROBID is a Local Variable.
- Variable %%\PROBID is a Global Variable.

■ In the IOA Global Variable database, the format for indicating a full path is as follows:

%%\product\application\group\job\variablename

■ Two variables with the same name but different paths are different variables. (This is comparable to the fact that two Unix or DOS files with the same name but different paths are different files. For example, File A under directory \A\B\C is a different file than File A under directory \D\E\F.)

#### **Example**

Due to the different paths, the following variables are all different from each other:

```
%%\M\APP_1\GRP_1\JOB_A\VAR_XYZ
%%\M\APP_1\GRP_1\JOB_B\VAR_XYZ
%%\M\APP_1\GRP_1\VAR_XYZ
```

- If the particular path has no Group and/or no Application (for example, the job does not belong to a group or application), CONTROL-M utilizes the keyword values "NO_APPL" and "NO_GROUP" in the path, as needed.
- Paths can be specified using the same rules and shortcuts that are available with directories and subdirectories (instead of the full path):
  - A job or Group Entity can assign a Global variable to itself by specifying a slash immediately following the %% symbol.

#### **Example**

If job JOB1 belongs to group GRP_A, which belongs to application APP_1, then the following SET VAR statement in JOB1:

```
SFT VAR=%%\PROBID=123
```

creates the following variable assigned to JOB1 (with the indicated full path):

```
%%\M\APP_1\GRP_A\JOB1\PROBID=123
```

— Paired dots with a backslash (..\) indicate movement to the next level up.

#### **Example**

If JOB1 belongs to group GRP_A, which belongs to application APP_1, then the following SET VAR statement in JOB1:

```
SET VAR=%%..\PROBID=123
```

creates the following variable assigned to GRP_A (with the indicated full path):

```
%%\M\APP 1\GRP A\PROBID=123
```

— To move directly down the hierarchy, it is only necessary to indicate the levels that are lower than the current level. (However, since only Group entities and jobs utilize variables, only Group entities can move directly down a level.

#### **Example**

Assume Group entity GRP_A wants to assign variable %%A, with a value equal to 7, (%%A = 7) to job JOB2. The following statement indicates the syntax of the %% SET statement:

```
//* %%SET %%\J0B2\A=7
```

- To move across the hierarchy (that is, to change paths), you can either:
  - Specify a full path, or
  - Move up to a component common to both paths, and then move down the other path.

#### **Example 1**

Assume job JOB1 in group GRP_A wants to assign variable %%A (with a value equal to 7) to job JOB2 in the same group (and assume the group has no application). Any of the following statements can be specified.

```
//* %%SET %%..\J0B2\A=7
//* %%SET %%\M\NO APPL\GRP A\J0B2\A=7
```

#### Example 2

Assume job JOB1, in group GRP_A, which is in application APP_A, wants to assign variable %%A, with a value equal to 7, to job JOB2, which is in group GRP B and which does not have an application.

Either of the following %% SET statements work:

```
//* %%SET %%\M\NO_APPL\GRP_B\JOB2\A=7
//* %%SET %%..\..\NO_APPL\GRP_B\JOB2\A=7
```

■ If two statements assign the same Global variable to the same component, the later assignment overwrites the earlier assignment.

#### **Example**

Assume job JOB1 belonging to group GRP_A is run before job JOB2 belonging to the same group (and both belong to application APP_A).

If JOB1 contains the following SET VAR statement:

```
SET VAR=\%..\A=7
```

and JOB2 contains the following SET VAR statement:

```
SET VAR=%%..\A=8
```

At the end of the job run for JOB2, the IOA Global Variable database contains the following variable (assigned to GRP_A):

```
%%\MAPP A\GRP A\A=8
```

■ A job or Group entity can utilize a Global variable that has been assigned to it by merely specifying the variable with the backslash, even if the variable was created by a different job. (The full path is not required.)

#### **Example**

Assume job JOB1 contained the following statement that assigned the following variable to JOB2.

```
DO SET VAR=%%\M\NO APPL\GRP A\JOB2\PROBID=123
```

JOB2 can then access this variable in a DO SHOUT statement without a full path name by specifying the variable with a backslash.

```
DO SHOUT TO TSO-U0014 URGENCY U

=*** Problem Encountered. Problem ID=%%\PROBID ***
```

- When changing paths or assigning a variable to a higher level component on the same path, a security check can be required.
- A job or Group entity can utilize a Global variable that has been assigned to a different component by specifying the appropriate path. However, before the variable could be utilized, security checks, if any, would have to be passed.

#### Example

If a Global variable is assigned to job JOB1 as in the following statement:

```
DO SET VAR=%%\M\NO APPL\GRP A\JOB1\PROBID=123
```

and JOB2 wants to access the variable, it would have to specify the appropriate path name (and pass any required security checks). For instance, JOB2 can access this variable in a DO SHOUT statement as follows:

```
DO SHOUT TO TSO-U0014 URGENCY U =**Problem Occurred. ID=%%\M\NO_APPL\GRP_A\JOB1\PROBID**
```

# **JCL Setup Operation Flow**

All JCL setup operations are performed during job submission. At this time, CONTROL-M processes the JCL of the job line by line.

CONTROL-M scans each line for AutoEdit terms (identified by the %% symbol) and tries to resolve them (unless otherwise instructed). CONTROL-M resolves all AutoEdit terms in a line before it moves to the next line.

All changes made during JCL processing (such as variable resolution) are retained only until CONTROL-M has finished submission of the job.

CONTROL-M resolves System variables by taking the values from the system.

CONTROL-M resolves Global variables by taking the values from the IOA Global Variable database.

Values for Local user-defined variables can be taken from any of several possible sources (described below). When CONTROL-M detects a local user-defined variable in the JCL line being processed, it checks these possible sources in a specific order until a value is found for the variable. CONTROL-M creates a user-defined variable environment in which it places each user-defined variable and its value.

The potential sources for local user-defined variable values are listed below in the order in which they are generally checked:

- System variable values
- %%SET control statements

These statements can be specified in JCL lines, including JCL comment lines. They assign values to variables.

■ SET VAR and DO SET statements

These statements can be specified in the job scheduling definition. They can be used to define new variables, or to assign new values to existing variables.

SET VAR statements can affect all job submissions.

DO SET statements can override values specified by a SET VAR or previous DO SET statement. However, since DO SET statements are post-processing parameters, they only affect subsequent runs of a job (rerun and restarted jobs).

 Local variables and values defined in members specified in %%LIBSYM / %%MEMSYM control statements

These members define local variables and specify their values. Members are searched in the order in which they appear in the JCL.

■ Local variables and values defined in members specified in %%GLOBAL control statements

These members define local variables and specify their values. Members are searched in the order they appear in the JCL.

The order in which CONTROL-M checks potential sources for possible AutoEdit variable resolution is important because once CONTROL-M has resolved a variable, it generally stops checking other sources. Potential values from other sources are ignored, and resolved values are not overridden except by %%SET statements in subsequent JCL lines.

Because JCL is processed sequentially one line at a time, the line being processed can only be affected by external members and %%SET control statements that have previously been processed. If a line contains an undefined variable that is only defined in a subsequent line, the variable cannot be resolved.

By default, if CONTROL-M cannot resolve a variable, it stops submission of the job. This default, however, can be overridden by specifying the %%RESOLVE control statement with a value of NO or OFF (described later in this chapter).

To stop submission of a job because of an unresolved variable, CONTROL-M creates an intentional JCL error that prevents execution of the job's already submitted JCL. The job ends with the status NOT SUBMITTED for reason JNSUB. The erroneous JCL remains on the spool, but does not affect other job executions except those that depend on the successful execution of this job.

Local variable values taken from variable members (%%MEMSYM and %%GLOBAL members) that are changed during job submission remain in effect only until CONTROL-M finishes submission of the job. Therefore, a change made to such a variable (using the %%SET control statement) affects only submission of that job and does not affect any other job submission or the value of the variable in the variable member.

# **Rules of Variable Resolution**

By default, columns 1 through 72 of JCL lines are searched for variables which are then analyzed and resolved. If column 72 contains an asterisk (*), the active range for resolution is columns 1 through 71 (to support continuation lines).

Multiple AutoEdit variables (and constants) can be joined together into a complex term. When a term contains multiple variables, those variables are resolved from right to left.

The methods of joining multiple variables together are described below.

■ Two variables can be joined to form a single complex variable by linking them together (such as %%A%%B).

### **Example 1**

Given:	%%A=1, %%B=2, %%A2=100	
Resolve:	%%A%%B	
Explanation:	The process of resolution is as follows:	
	Initial expression to resolve	%% <b>A</b> %%B
	Resolve %%B	2
	Replace %%B with value 2	%%A2
	(%%A%%B partially resolves to a single variable %%	A2)
	Resolve %%A2	100
Solution:	%%A%%B resolves to 100	

#### **Example 2**

Given:	The day is the 3rd of the month.
Resolve:	//SYSBKP DD UNIT=TAPE, VOL=SER=%%BACKUP_TAPE_%%ODAY,
Solution:	This statement partially resolves to:
	//SYSBKP DD UNIT=TAPE, VOL=SER=%%BACKUP_TAPE_03,
	%%BACKUP_TAPE_03 is a single user-defined variable. If the value of this variable is known to CONTROL-M as EE1022, the statement would fully resolve to:
	//SYSBKP DD UNIT=TAPE,VOL=SER=EE1022,

■ Two variables can be concatenated into two distinct but joined variables by placing a period between them (such as %%A.%%B).

### **Example 1**

Given:	%%A=1, %%B=2, %%A2=100	
Resolve:	%%A.%%B	
Explanation:	The process of resolution is as follows:	
	Initial expression to resolve	%%A.%%B
	Resolve %%B	2
	(The partially resolved variable now reads	%%A.2)
	Resolve %%A	1
	(The partially resolved variable now reads	1.2)
	Final resolution of the two values (based on the rule that two variables joined by a period resolve to a concatenated value)	12
Solution:	%%A.%%B resolves to 12	

### **Example 2**

On the 4th of December, %%ODAY.%%OMONTH resolves to 0412

(If the expression had been written %%ODAY%%OMONTH (without the period), it would have partially resolved to %%ODAY12, which is a user-defined variable requiring further resolution.)

■ Two variables can be concatenated into two distinct variables joined by a period by placing two periods between them (such as %%A..%%B).

### **Example 1**

Given:	%%A=1, %%B=2, %%A2=100	
Resolve:	%%A%%B	
Explanation:	The process of resolution is similar to the resolution of two variables joined by one period:	
	Initial expression to resolve	%%A%%B
	Resolve %%B	2
	(The partially resolved variable now reads	1.2)
	Resolve %%A	1
	(The partially resolved variable now reads	12)
	Final resolution of the two values (based on the rule that two variables joined by two periods resolve to two values joined by a period)	1.2
Solution:	%%A%%B resolves to 1.2	

### **Example 2**

On the 4th of December, %%ODAY..%%OMONTH resolves to 04.12

■ A constant can be appended to a variable by prefixing the constant with the concatenation symbol %%. For example, in expression %%AA%%.UP, constant UP is appended to variable %%AA.

Without symbol %%., the constant would be treated as part of the variable (for example, expression %%AAUP consists of one variable).

The %%. symbol is not required if the constant precedes the variable (for example, UNIT%%AA) since the %% prefix of the variable differentiates it from the constant.

### **Example**

Given:	%%MODE = PROD			
Resolve:	CTM.%%MODE%%.01.JCL			
Explanation:	The process of resolution is as follows:			
	Initial expression to resolve	CTM.%%MODE%%.01.JCL		
	Resolve %%MODE	PROD		
	(The partially resolved variable now reads CTM.PROD%%.01.JCL)			
	Final resolution (based on the rule that symbol %%. joins a constant to a variable)	CTM.PROD01.JCL		
Solution:	CTM.%%MODE%%.01.JCL resolves to CTM.PROD01.JCL			

#### - NOTE -



To separate a constant (JCL) from a variable (%%MODE) by a period, specifying the period is sufficient. For example: CTM.%%MODE.JCL would resolve to CTM.PROD.JCL.

# **Order of Precedence for Multiple Value Assignments**

If a particular AutoEdit variable can receive a value from more than one source, an order of precedence is necessary to determine which of the possible values is assigned.

The following chart indicates the order of precedence. The chart works as follows:

■ Each row in the chart represents a possible source of a value for a variable.

- In each column, a single pair of value sources (rows) are selected and compared for precedence:
  - The source that takes precedence in the pair is identified by label P.
  - The other source in the pair is identified by label O.

When many sources of value assignments are available for a variable, use the chart below to compare those sources one pair at a time, as follows:

- From the list of available sources for the particular variable, select any two sources and use the chart to eliminate the source of lower priority. The list now has one less source available.
- Repeat this process until only the source of highest priority remains.

**Table 245 Chart for Determining Priorities of Value Assignment Sources** 

Source of Value Assignment		Paired Combinations of Value Sources							
SET VAR (Job Scheduling Definition)	O	P	P						
JCL SET (earlier)	P			P	P		0		
JCL SET (later)							P		
LIBSYM		О		0		P		P	
LIBSYM2								О	
GLOBAL			О		О	О			P
GLOBAL2									О

#### NOTE



JCL SET can apply to an actual AutoEdit SET statement in the JCL or to AutoEdit SET statements embedded within an INCLIB member referenced in the JCL.

LIBSYM represents a member specified in an earlier statement; LIBSYM2 represents a different member specified in a later statement. The same applies to GLOBAL and GLOBAL2 respectively.

If there are multiple value assignments for the same variable in the one member, the last assignment in that member is used for the above calculation.

# **Control Statements**

Control statements define the AutoEdit environment and control AutoEdit processing.

Control statements can appear anywhere in the JCL member to be submitted. When a control statement is detected in a JCL line (for example, in a JCL remark statement), the line containing the control statement is submitted as part of the job. If the control statement appears in a non-JCL line (for example, a line beginning without a // symbol), the control statement is resolved and the resolved value can be applied to subsequent JCL lines, but the control statement is not submitted as part of the job.

Available control statements are discussed on the following pages.

## %%GLOBAL

Control statement %%GLOBAL defines a member that contains a set of user-defined variables and values. Before job submission, the CONTROL-M monitor reads this member from the library referenced by DD statement DAGLOBAL in the CONTROL-M procedure. The content of the member is added to the user-defined variable environment of the job.

You can specify more than one %%GLOBAL statement for a job. Each statement is added to the user-defined variable environment of the job.

Global members can also be placed in cache, which can significantly improve performance if the member is used by many jobs. For more information, see "Loading %%GLOBAL Members to Cache" on page 758, and the corresponding topic in the *INCONTROL for z/OS Administrator Guide*.

If CONTROL-M fails to access the variable member (member not found, and so on), the job is not submitted and a warning message is issued to the user who requested the job.

The format of the %%GLOBAL control statement is:

```
//* %%GLOBAL memname
```

where *memname* is a valid member name of 1 through 8 characters.

#### **Example**

```
//* %%GLOBAL TAPES
//* %%GLOBAL CURRENCY
```

## %%GOTO and %%LABEL

%%GOTO and %%LABEL control statements provide the AutoEdit facility with "GO TO" logic, permitting simple inclusion or exclusion of job steps, DD statements, input date, and so on.

The format of %%GOTO and %%LABEL statements is:

```
%%GOTO labelname
%%LABEL labelname
```

The %%GOTO statement transfers control to the location in the program designated by a matching %%LABEL statement. The search for a matching %%LABEL *labelname* is only performed downward (that is, loops are not supported).

All statements between a %%GOTO statement and its matching %%LABEL statement are not processed (that is, no statements are submitted and AutoEdit statements are not resolved).

%%GOTO and %%LABEL statements are generally used in conjunction with %%IF, %%ELSE, and %%ENDIF control statements. Examples at the end of this chapter demonstrate how these statements can be combined.

## %%IF, %%ELSE, %%ENDIF

%%IF, %%ELSE, and %%ENDIF control statements provide the AutoEdit facility with Boolean "IF" logic capability. These statements, in conjunction with %%GOTO and %%LABEL control statements, permit branching based on submission time criteria. Job steps, DD statements, and so on are easily excluded or included.

The format of %%IF, %%ELSE, %%ENDIF statements is:



where:

■ conditional-expression is the condition that determines whether the accompanying statements are performed. If the condition is satisfied, the statements accompanying the %%IF statement are performed and the statements accompanying the %%ELSE statement (if specified) are not performed. If the condition is not satisfied, the statements accompanying the %%ELSE statement (if specified) are performed and the statements accompanying the %%IF statement are not performed.

The format of a conditional-expression is: operand operator operand

#### where:

- operand Any character string. It can contain AutoEdit terms.
- *operator* One of the valid comparison operators listed below. Valid operators:
  - EQ is equal to
  - NE is not equal to
  - GT is greater than
  - GE is greater than or equal to
  - LT is less than
  - LE is less than or equal to
- statements are any statements specified in the JCL member, including AutoEdit statements, JCL statements and non-JCL statements.

If an operand contains AutoEdit terms, they are resolved into a character string before the conditional expression is analyzed.

An operand must not resolve to a null value (as in CLISTs). If it is possible that an operand resolves to a null value, place a character before the first and second operands in a way that would not affect the comparison. For example, if %%A and/or %%C in the statement:

```
%%IF %%A GT %%C
```

might resolve to null, use a substitute expression such as:

```
%%IF B%%A GT B%%C
```

Operands are compared as character strings from left to right. For example:

91 is greater than 1000 (because 9 is greater than 1).

An %%IF expression must be terminated with an %%ENDIF statement. If an %%IF expression is not terminated in this way, an %%ENDIF statement is implied as the last statement in the member.

The %%ELSE statement is optional.

When the %%IF expression is true, all JCL statements (including non-AutoEdit statements) between the %%IF expression and its %%ELSE statement (or its matching %%ENDIF statement when no %%ELSE statement is present) are submitted by CONTROL-M provided that all AutoEdit variables are resolved.

When the %%IF expression is not true and an %%ELSE statement exists, only statements between the %%ELSE statement and the matching %%ENDIF statement are submitted.

%%IF statements can be nested.

The following logically connected statements may be specified on one line:

- %%IF
- %%ELSE
- %%ENDIF
- %%GOTO
- %%SET

## **Example**

Up to 100 nested %%IF statements can be specified.

## %%INCLIB and %%INCMEM

%%INCLIB and %%INCMEM statements contain two elements that together describe the member that is to be included in the current job stream, as follows:

- The %%INCLIB part of the statement defines the location of the member as one of the following:
  - the library name
  - the DD name to be associated with a library or concatenation of libraries
- The %%INCMEM part of the statement defines the member.

These statements are useful for inserting the following types of information into the JCL:

- JCL statements and parameters to be passed to the JCL (for example, sysin)
- AutoEdit control statements, including other %%INCLIB and %%INCMEM statements

The format of the statement is:

```
%%INCLIB { libname | DDNAME=ddname } %%INCMEM memname
```

#### In this statement

- *libname* is a valid data set name, from 1 through 44 characters in length, of a cataloged partitioned data set (library) with a record length of 80
- ddname is a valid DD name from 1 through 8 characters in length that points to a cataloged library or concatenation of cataloged libraries
  This DD name must be preallocated to the environment in which the %%INCLIB statement is to be resolved, such as the CONTROL-M monitor or the IOA online logon procedures.
- *memname* is a valid member name from 1 through 8 characters in length

You can specify multiple %%INCLIB and %%INCMEM statements in a job.

More than one job may contain identical %%INCLIB and %%INCMEM statements, permitting maintenance of common, standardized code.

The %%INCMEM member is read by the CONTROL-M monitor just before job submission, and the contents of the member are submitted as part of the current job. As a result

- a member created by one job in the job stream can be used by a later job in the job stream
- if a job in the job stream updates a member and the member is subsequently used by a later job in the job stream, the later job accesses the updated member

If the %%INCLIB statement is resolved within the JCL, ensure that there are no unnecessary blank lines in the %%INCMEM member.

If CONTROL-M fails to access the included member (member not found, and so on), the job is not submitted and a warning message is issued.

## %%LIBSYM and %%MEMSYM

Control statements %%LIBSYM and %%MEMSYM define a library and a member that contain a set of user-defined variables and their assigned values. The member is read by CONTROL-M before submission, and its content is added to the user-defined variable environment of the job.

It is possible to specify more than one %%LIBSYM or %%MEMSYM statement for one job. Each statement is added to or deleted from the user-defined variable environment of the specific job.

If CONTROL-M fails to access the variable member (member not found, security constraints, and so on), the job is not submitted and a warning message is issued to the user who requested the job.

The format of the statement is:

%%LIBSYM *libname* %%MEMSYM [-]memname

#### where:

- *libname* Valid data set name of 1 through 44 characters. It must be a cataloged partitioned data set (library) with a record length of 80.
- *memname* Valid member name of 1 through 8 characters.

When a minus sign (-) is specified before *memname*, the purpose of the %%LIBSYM/%%MEMSYM statement is to cancel the effect of the previously processed %%LIBSYM *libname* %%MEMSYM *memname* statement with the same member name, thereby reverting the values of the AutoEdit variables that are defined in the LIBSYM/MEMSYM member to the values that were in effect prior to the previous %%LIBSYM *libname* %%MEMSYM *memname* statement. This, in effect, provides the capability of creating local auto-edit variables, whose scope is restricted to specified range(s) of the job run stream.

### %%RANGE

A %%RANGE statement limits the handling of AutoEdit functions and variables to a specified column range. The contents of all columns outside the range remain unchanged.

This statement is useful when values must be specified in specific columns and when not every AutoEdit statement need be resolved.

The format of the statement is:

%%RANGE fromcol tocol

#### where:

- *fromcol* First column in the range. Valid values are: 1 through 80. The default (without a range statement) is 1.
- *tocol* Last column in the range. Valid values are: 1 through 80. *tocol* must be a value equal to or greater than *fromcol*. The default (without a range statement) is 72.

#### NOTE



When used in a CONTROL-M SETVAR job definition, the format of the %%RANGE statement should be:

%%RANGE=fromcol,tocol

The %%RANGE statement can prevent the shifting of constants and variables that appear after an AutoEdit variable in the same line. By limiting AutoEdit resolution to a specified range, all constants and variables outside the specified range are kept in their original positions regardless of the length of the resolved variables.

Each %%RANGE statement is valid until a new %%RANGE statement is specified. Note, however, that the placement of the subsequent %%RANGE statement must be within the column range of the preceding %%RANGE statement (or it is not recognized as an AutoEdit statement).

#### NOTE



The minimum length of a %%RANGE statement with 2-digit fromcol and tocol values is 12 characters. Do not, therefore, specify a range of fewer than 12 columns, or you cannot use a subsequent %%RANGE statement to expand the range back to the regular line length.

## **Example**

This example shows how a %%RANGE statement affects the resolution of a line. In the original JCL, the %%RANGE statement affects the second occurrence of the AutoEdit variable, but not the first. In the submitted JCL, note the impact on the positioning of constant CONSTANT.

#### The original JCL:

```
//* %%SET %%A_VERY_LONG_VARIABLE=XXX

%%A_VERY_LONG_VARIABLE CONSTANT

//* %%RANGE 1 25

%%A_VERY_LONG_VARIABLE CONSTANT
```

#### The submitted JCL:

```
//* %%SET %%A_VERY_LONG_VARIABLE=XXX

XXX CONSTANT
//* %%RANGE 1 25

XXX CONSTANT
```

## %%RESOLVE

By default, CONTROL-M must resolve all AutoEdit terms in the JCL or the job is not submitted. This default can be overridden by specifying an appropriate %%RESOLVE statement in the JCL.

Valid %%RESOLVE statements are:

Table 246 %%RESOLVE Statements (part 1 of 2)

Statement	Description
%%RESOLVE NO	Try to resolve AutoEdit terms. If an AutoEdit term cannot be resolved, submit the job with the AutoEdit term as is.
%%RESOLVE YES	If YES, MUST or blank is specified and a subsequent AutoEdit term cannot be resolved, the job is not submitted.
%%RESOLVE MUST	
%%RESOLVE (blank)	
%%RESOLVE OFF	Do not try to resolve AutoEdit terms except for other
	%%RESOLVE statements. Submit the job with AutoEdit terms as
	is.
Each %%RESOLVE staten	nent remains in effect until the next %%RESOLVE statement in the
JCL is encountered.	

Table 246 %%RESOLVE Statements (part 2 of 2)

### Statement Description The following special case %%RESOLVE statement is relevant if %%GLOBAL AutoEdit members are loaded to cache. %%RESOLVE If an AutoEdit variable has not been resolved by searching the ALLCACHE {OFF | ON} %%GLOBAL members identified in the job, the %%RESOLVE ALLCACHE statement instructs CONTROL-M to continue the variable resolution process by checking all members loaded into cache. Members in cache are searched in the same sequence they are listed in the cache list member. A %%RESOLVE ALLCACHE statement without an ON or OFF qualifier can only be specified as the first non-comment statement in the cache list member used to load %%GLOBAL members to cache. For more information, see "Loading %%GLOBAL Members to Cache" on page 758. A %%RESOLVE ALLCACHE statement with an ON or OFF qualifier can be specified anywhere in the JCL of the job. It overrides the most current %%RESOLVE ALLCACHE function, as follows: %%RESOLVE ALLCACHE ON - Activates the %%RESOLVE ALLCACHE function. %% RESOLVE ALLCACHE OFF - Deactivates the %%RESOLVE ALLCACHE function.

#### NOTE



When used in a CONTROL-M SETVAR job definition, the format of the %%RESOLVE statement should be:

%%RESOLVE=value

## **%%SET**

A %%SET control statement sets the values of user-defined variables. The statement may be placed in any part of the JCL stream.

The format of the statement is:

%%SET %%varname=expression

#### where:

- *varname* a valid user-defined variable
- expression must resolve to a value according to the rules described in "Rules of Variable Resolution" earlier in this chapter or submission of the job is canceled (unless a %%RESOLVE NO control statement is specified). An expression can consist of a:
  - value (for example, 5)
  - variable (for example, %%ODATE)
  - a combination of values, variables, operators, functions, and so on (for example, %%GENERATION_NUMBER %%PLUS 1).

### **Example 1**

```
%%SET %%BACKUP_UNIT=TAPE
```

User-defined variable %%BACKUP_UNIT is assigned the value TAPE.

### **Example 2**

```
%%SET %%BACKUP_UNIT_%%WDAY=EE%%OMONTH.%%ODAY
```

On Monday the 24th of September, user-defined variable %%BACKUP_UNIT_1 is assigned the value EE0924.

### **Example 3**

```
//* %%SET %%SCRATCH=%%SCRATCH %%PLUS 1
//SYSUT1 DD UNIT=TAPE.VOL=SER=EE%%SCRATCH.DISP=...
```

When the initial value of SCRATCH is 3017, the result in the submitted member is:

```
//* %%SET %%SCRATCH=3017 %%PLUS 1
//SYSUT1 DD UNIT=TAPE,VOL=SER=EE3018,DISP=...
```

# **Operators**

AutoEdit operators are used to add or subtract values from AutoEdit variables in the JCL. These operators can only be specified in a %%SET statement. Valid AutoEdit operators are:

#### **Table 247 AutoEdit Operators**

Operator	Description
%%PLUS	Adds a value to an AutoEdit variable.
%%MINUS	Subtracts a value from an AutoEdit variable.

#### AutoEdit operators are generally used as follows:

%%SET variable=operand operator operand

#### where:

- *operand* Expression that resolves to a numeric value.
- *operator* %%PLUS or %%MINUS.

Only one operator can be specified in each %%SET statement.

## **Example**

Increase the number of generations (%%GENERATION_NUMBER) by one:

// %%SET %%GENERATION_NUMBER=%%GENERATION_NUMBER %%PLUS 1

If the value of %%GENERATION_NUMBER was initially 1, it is set to 2.

# **Functions**

AutoEdit functions perform operations on specified AutoEdit variables in the JCL. These functions can only be specified in %%SET statements. The following AutoEdit functions are supported by CONTROL-M:

## %%\$CALCDTE

The %%\$CALCDTE function performs date manipulation by adding or subtracting a specified number of days from a specified date.

#### NOTE



This function replaces the old %%CALCDATE function, which is still supported for backward compatibility. BMC Software recommends that you use the %%\$CALCDTE function rather than the %%CALCDATE function, to take advantage of its increased versatility.

The format of the %%\$CALCDTE function is:

%%\$CALCDTE date ± [quantity_type]quantity

#### where:

- *date* must be (or resolve to) a date in format yyyymmdd
- quantity_type optional 1-character description of the type of data specified as quantity

Valid values are:

- D days
- M months
- Y years

If no value is specified, the default value is D (days).

quantity – number (or numeric AutoEdit expression) of date units, depending on the value specified for quantity_type to add to or subtract from the date Valid values are: 1 through 999.

#### NOTE -



In setting values for the *quantity_type* and *quantity* variables, ensure that the final date is not later than the year 2054.

### **Example 1**

```
//* %%SET %%A=%%$CALCDTE %%$ODATE -1
```

If the original scheduling date is February 1, 2001, %%A is assigned a value of 20010131.

#### **Example 2**

```
//* %%SET %%A=%%$CALCDTE %%$ODATE +M1
```

If the original scheduling date is January 30, 2002, %%A is assigned a value of 20020228.

#### Example 3

```
//* %%SET %%A=%%$CALCDTE %%$ODATE +Y1
```

If the original scheduling date is February 29, 2000, %%A is assigned a value of 20010228.

#### NOTE



If as a result of adding months to a date, the number of days exceeds the maximum number of days possible in the resulting month, CONTROL-M reduces the number of days to the actual maximum.

## %%\$GREG

The %%\$GREG function converts a Julian date (with a 4-character year) to a Gregorian date (with a 4-character year). The format of function %%\$GREG is:

%%\$GREG date

where date must be (or resolve to) a date in Julian format yyyyddd.

### **Example**

//* %%SET %%A=%%\$GREG 2001196

%%A is assigned a value of 20010714

## %%\$JULIAN

The %%\$JULIAN function converts a Gregorian date (with a 4-character year) to a Julian date (with a 4-character year). The format of the %%\$JULIAN function is:

%%\$JULIAN date

where *date* must be (or resolve to) a date in format *yyyymmdd*.

#### **Example**

//* %%SET %%A=%%\$JULIAN 20010717

%%A is assigned a value of 2001197

## **%%\$LEAP**

The %%\$LEAP function determines whether a specified Gregorian date (with a 4-character year) falls in a leap year. If the date is in a leap year, the variable resolves to 1. If the date is not in a leap year, the variable resolves to 0. The format of the %%\$LEAP function is:

%%\$LEAP date

where date must be (or resolve to) a date in format yyyymmdd.

Leap years are years whose last two digits are evenly divisible by 4, excluding those years that are divisible by 100 but not by 400. Therefore, 2000 is a leap year but the years 2100, 2200 and 2300 are not.

### **Example**

//* %%SET %%A=%%\$LEAP %%\$ODATE

%%A is assigned a value of 1 for dates in the year 2000 and 0 for dates in the year 2001.

## %%\$WCALC

The %%\$WCALC function performs a shift from the specified date to a working date in the specified calendar, according to indicated instructions. The format of the %%\$WCALC function is:

%%\$WCALC date instruction calendar

#### where:

- *date* must be (or resolve to) a date in format yyyymmdd
- *instruction* shift instructions.

#### Valid values are:

- -+n Shift to the next *n*th working date in the calendar.
- --n Shift to the previous *n*th working date in the calendar.
- > If the specified date is not a current working date, shift to the next working date in the calendar. (If the specified date is a working date, do not shift.)
- < If the specified date is not a current working date, shift to the previous working date in the calendar. (If the specified date is a working date, do not shift.)
- calendar name of the calendar to check for working dates

### **Example**

```
//* %%SET %%A=%%$WCALC 20000717 +1 EXCPTCAL
//* %%SET %%A=%%$WCALC 20000717 > EXCPTCAL
```

- If calendar EXCPTCAL (for 2000) contains consecutive working dates 07/13 and 07/20, %%A resolves to 20000720 in both %%SET statements.
- If calendar EXCPTCAL (for 2000) contains consecutive working dates 07/17 and 07/24:
  - In the first %%SET statement (with the +1), %%A resolves to 20000724.
  - In the second %%SET statement (with the >), %%A resolves to 20000717.

## %%**\$WEEK**#

The %%\$WEEK# function calculates in which week of the year (1 through 53) a specified date falls. The function uses the site-defined start of the week (Sunday or Monday) as the first day of each week, and assumes that January 1st falls in the first week.

This function ensures that every day of the year falls into a week of that year, but it also means that the first week of the year may possibly have a majority of its days come from December of the preceding year.

(By contrast, the %%\$YEARWK# AutoEdit function, which also calculates in which week of a year a date falls, counts the week that includes January 4th as the first week. This ensures that the first week in the year has a majority of its days in January. However, it also means that the first days of the year may possibly belong to the last week of the preceding year, and the last days of the year may possibly belong to the first week of the following year.)

The format of the %%\$WEEK# function is:

```
%%$WEEK# date
```

where *date* is the date in format yyyymmdd (a 4-character year must be specified).

### **Example**

```
//* %%SET %%A=%%$WEEK# 20010712
```

%%A is assigned a value of 28

## %%\$WEEKDAY

The %%\$WEEKDAY function calculates on which day of the week a specified date (with a 4-character year) falls. The resolved value is an integer from 1 through 6 or 0, where 1 corresponds to the first day of the week (Sunday or Monday, depending on the site-standard) and 0 corresponds to the last day of the week (Saturday or Sunday).

The format of the %%\$WEEKDAY function is:

```
%%$WEEKDAY date
```

where date must be (or resolve to) a date in format yyyymmdd.

#### **Example**

```
//* %%SET %%A=%%$WEEKDAY 20000714
```

%%A is assigned a value of 6 (Friday)

# %%\$YEARWK#

The %%\$YEARWK# function calculates in which week of the year (1 through 53) a specified date falls, and returns the year and the week number according to ISO8601 standards. In accordance with those standards, the function uses Monday as the first day of each week (this is so even if the start of the week at your site is defined as Sunday).

The %%\$YEARWK# function assumes that the first week is the week that includes January 4th

This function ensures that the first week in the year has a majority of its days in January. However, it also means that the first days of the year may possibly belong to the last week of the preceding year, and the last days of the year may possibly belong to the first week of the following year.

By contrast, the %%\$WEEK# AutoEdit function, which also calculates in which week of a year a date falls, counts the week that includes January 1st as the first week. This ensures that every day of the year is part of a week of that year. However, it also means that the first week of the year may possibly have a majority of its days in December of the preceding year.

The format of the %%\$YEARWK# function is:

```
%%$YEARWK# date
```

where date is the date in format yyyymmdd (a 4-character year must be specified).

The value returned by the function is in the format:

yyyyWnn

where:

- *yyyy* year in which the week falls
- *nn* number of the week within the year

#### **Example 1**

```
//* %%SET %%A=%%$YEARWK# 20010214
```

%%A is assigned a value of 2001W06

#### Example 2

```
//* %%SET %%A=%%$YEARWK# 20050101
```

%%A is assigned a value of 2004W52

#### **Example 3**

```
//* %%SET %%A=%%$YEARWK# 20011231
```

%%A is assigned a value of 2002W01.

# %%CALCDATE

The %%CALCDATE function performs date manipulation by adding or subtracting a specified number of days from a specified date.

#### NOTE



The %%CALCDATE function is supported for backward compatibility. BMC Software recommends that you use the %%\$CALCDTE function instead.

The format of the %%CALCDATE function is:

%%CALCDATE date ± quantity

#### where:

- *date* must be (or resolve to) a date in format yymmdd
- quantity number (or numeric AutoEdit expression) of days (from 1 to 366) to add to or subtract from the date

## %%SUBSTR

The %%SUBSTR function extracts a substring from a string.

The format of the %%SUBSTR function is

%%SUBSTR string startpos length

#### where

- string string from which the substring is extracted
- *startpos* character position in the original string from which the extraction begins
- *length* number of characters to extract

A new string is created composed of the characters extracted from the original string.

startpos and length must be a numeric value or AutoEdit expression that is greater than zero.

When the starting position of the substring is greater than the argument *string*, the function returns a null value.

When the starting position of the substring falls within the argument *string*, but the length of the substring falls outside the range of the argument *string* (*startpos* + *length* – 1), the function returns a substring containing the characters from the starting position.

If the character positions of *startpos* + *length* – 1 is greater than the *string* length, submission of the member is stopped.

### **Example 1**

```
//* %%SET %%A=%%$CALCDTE %%$ODATE -1
//* %%SET %%AMON=%%SUBSTR %%A 5 2
```

#### On July 1, 2001:

%%A is assigned a value of 20010630

%%AMON is assigned a value of 06

### **Example 2**

```
%%SET %%A=%%SUBSTR CABLE 4 4
```

resolves to

%%A=LE

## %%\$LENGTH

The %%\$LENGTH function returns the length of a character string.

The format of the %%\$LENGTH function is

```
%%$LENGTH char_string
```

where *char_string* is, or resolves to, any character string.

### **Example**

```
//* %%SET %%A=%%$LENGTH A123
```

%%A is assigned a value of 4.

# **%%\$TYPE**

The %%\$TYPE function returns the type attribute of a character string. Possible type attributes are:

- N numeric
- M negative numeric
- C character
- X alphanumeric
- 0 undefined or 0 length

The format of the %%\$TYPE function is

```
%%$TYPE char_string
```

where char_string is, or resolves to, any character string.

### **Example 1**

```
//* %%SET %%A=%%$TYPE A123
```

%%A is assigned a value of X

### **Example 2**

```
//* %%SET %%B=%%$TYPE XYZ
```

%%B is assigned a value of C

### **Example 3**

```
//* %%SET %%C=%%$TYPE -1239
```

%%C is assigned a value of M

## %%\$FUNC

%%\$FUNC is an AutoEdit function that enables the creation of user-defined functions. You can only use the %%\$FUNC function as part of an AutoEdit %%SET statement.

The syntax for such use of the %%\$FUNC function is

```
%%SET output_char_string = %%$FUNC func_name input_char_string
```

In its operation, it is equivalent to using assembler language to issue the following CALL instruction

```
CALL func_name, (input_char_string, output_char_string)
```

#### In this instruction

- *func_name* is the name of the user-coded program module
- *input_char_string* is a string consisting of two parts in the following order:
  - the length of the source string
  - the source string
- *output_char_string* is a string consisting of two parts in the following order:
  - the length of the result string
  - the result string returned by func_name

The AutoEdit processor passes these parameters as variable-length strings. Each string consists of a half-word binary length field followed by the string itself. The *func_name* program must return the output string in the same format, as illustrated in the example below.

The source string can contain AutoEdit variables. If it does, these variables are resolved before the function is activated.

The maximum length of the source string, after resolving any AutoEdit variables, is 240 characters.

The maximum length of the result string is also 240 characters.

Neither the source string nor the result string can contain non-displayable characters.

You can use AutoEdit simulation to test your program module. For more information, see "M2: Perform an AutoEdit Simulation" on page 328.



#### NOTE -

You can define your *func_name* program module as resident. A resident program module is loaded once, kept in the storage, and entered by means of either the CALL instruction or a LINK instruction. If you want to do this, the program module must comply with both the following conditions:

- It must be able to work in AMODE 31.
- It must be reentrant.

To define program modules as resident, include them in the cache members list using the following definition syntax:

%%\$FUNC func_name

### **Example**

The user has a multiply function that is performed by a module named MULT.

The user's JCL contains the following AutoEdit statements:

```
%%SET %%A = 20
%%SET %%B = 30
%%SET %%C = %%$FUNC MULT %%A %%B
```

The last %%SET statement causes the CONTROL-M monitor to call the MULT module as follows (using assembler notation):

```
CALL MULT, (PRM1, PRM2)
```

The PRM1 and PRM2 parameters are passed to MULT in the following format:

The MULT program returns results by updating the value of the second parameter, PRM2, as follows:

```
PRM2 DC H'3'
DC C'600'
```

The result is that the AutoEdit variable %%C is assigned a character value of 600.

## **Testing AutoEdit Syntax**

When CONTROL-M detects an AutoEdit syntax error in a JCL member during submission, the submission is canceled by CONTROL-M. Therefore, it is essential to check the syntax of AutoEdit statements while the member is being prepared.

Furthermore, when the syntax is correct, you may want to verify that the AutoEdit statements return the desired results. For example, you may want to check that you specified the correct AutoEdit date variables for a job that performs end-of-year processing.

The CTMAESIM utility tests AutoEdit syntax and JCL setup. This utility simulates the actions of the CONTROL-M submission mechanism, which performs AutoEdit processing and JCL setup, and produces a printed report of the process.

CONTROL-M has a customized interface with the JOB/SCAN and PRO/JCL products. However, this utility can be used with any JCL-checking product.

The CTMAESIM utility can operate in either JCL Library mode or Scheduling Library mode:

- In JCL Library mode, the utility checks the AutoEdit statements in the job's JCL.
- In Scheduling Library mode, the utility not only checks the AutoEdit statements in the job's JCL of the job, it also checks the impact that SET VAR statements in the job scheduling definition have on the JCL of the job.

The utility enables system programmers to check the operation of CONTROL-M submission exit CTMX002 without affecting production.

The CTMAESIM utility can be activated by a batch procedure or the Online facility, as follows:

- Batch procedure using procedure CTMAESIM
- Online facility using option M2 in the Online Utilities menu or using CLIST CTMCAES

The utility requires specification of various parameter statements that determine how the simulation works, and which provide necessary information for the simulation.

Although the simulation works in much the same manner whether activated by a batch procedure or online, the following differences depend on the method of activation:

Batch activation allows specification of multiple sets of parameter statements.
 Online activation allows specification of only one set of parameter statements.

- Batch activation allows inclusion or omission of parameter RDR=INTRDR (described below). This parameter cannot be specified online.
- Command JSCAN (available only at sites where JOB/SCAN or PRO/JCL is installed) can only be specified if the utility is activated through the Online Utilities menu. It cannot be specified if the utility is activated by batch or by CLIST CTMCAES.
- Character masking is not supported in the Online utility. In the batch utility, character masking is supported for the member name in JCL Library mode, and for the job name in Scheduling Library mode. Valid mask characters are:
  - * Represents any number of characters (including no characters)
  - ? Represents any one character
- The SET VAR parameter, which can be specified outside the job scheduling definition, is supported in batch mode only. However, SET VAR statements in the job scheduling definition can be checked in both online and batch mode.
- The CONTROL-M GLOBAL LIBRARY parameter is specified only in the Online utility, and only one library can be specified. In batch mode, global libraries are referenced by the DAGLOBAL DD statement (multiple libraries can be concatenated).

In addition, depending on the command type specified in a parameter statement, the resulting JCL lines can also be written to the output file referenced by the DASUBMIT DD statement.

When the JCL is written to the output file referenced by the DASUBMIT DD statement, the output file can be routed to an internal reader by specifying the parameter RDR=INTRDR in the EXEC statement. In this case, the DASUBMIT DD statement is allocated as SYSOUT=(class,INTRDR) and the job is submitted.

Submission of the job enables the JCL to be checked by the JCL checking mechanism of MVS.

#### NOTE



A DASUBMIT DD statement can also be used by the AutoEdit Simulation facility to submit jobs for execution in emergency situations (for example, the CONTROL-M monitor is inoperative due to a severe technical problem).

When activated using ISPF, the functioning of the utility is similar to its functioning when activated from batch with the parameter RDR=INTRDR specified.

The CTMAESIM utility, as activated from the Online facility, is described in Chapter 2, "Online Facilities." The CTMAESIM utility, as activated through a batch procedure, is described in the *INCONTROL* for *z/OS Utilities Guide*.

# AutoEdit Usage in the Job Scheduling Definition

Certain AutoEdit components can be used in job scheduling definitions. In job scheduling definitions, AutoEdit components in certain statements (SET VAR and DO SET) directly affect JCL. In other statements (SYSOUT and DO SYSOUT, SHOUT and DO SHOUT, MEMLIB and OVERLIB) they do not affect the JCL.

#### WARNING



When using AutoEdit variables in job scheduling definitions, always test the definitions for proper AutoEdit syntax with the AutoEdit simulation utility. For more information, see "Testing AutoEdit Syntax" on page 793.

In the job scheduling definition, AutoEdit components can be specified in the following parameters:

■ SET VAR and DO SET statements

These two job scheduling definition statements and the %%SET control statements are used to assign values to user-defined variables in the JCL.

### **Example**

In this example, AutoEdit statements in the job scheduling definition and the JCL allocate space for the job. If the job abends due to insufficient space, the AutoEdit statements adjust the allocated space and rerun or restart the job.

The following step in the job's JCL sets the quantity of available space to five units of whatever type (track or cylinder) is specified in the job scheduling definition.

```
//STEP10 EXEC PGM=MYPGM
//OUTFILE DD DSN=NEWFILE,DISP=(NEW,CATLG,DELETE),
// SPACE=(%%SPACE_TYPE,5),UNIT=SYSDA
```

The job scheduling definition contains the following SET VAR statement that sets the space type to "track":

```
SET VAR %%SPACE_TYPE=TRK
```

In this case, the second line in the above DD statement resolves to:

// SPACE=(TRK,5),UNIT=SYSDA

The job scheduling definition also contains the following statements that are activated if the job abends due of lack of space (code S*37). These statements change the space type to "cylinder", which provides enough space, and rerun the job. If CONTROL-M/Restart is active, the job is restarted from the abended step.

```
ON STEP STEP10 CODES S*37

DO SET %%SPACE_TYPE=CYL

[DO IFRERUN FROM $ABEND] ===> If CONTROL-M/Restart is active

DO RERUN
```

If the job abends as above, the second line of the earlier JCL DD statement resolves to:

// SPACE=(CYL,5),UNIT=SYSDA

when the job is submitted for rerun (or restart).

SYSOUT and DO SYSOUT

File names for SYSOUT and DO SYSOUT handling can be specified using AutoEdit variables whenever SYSOUT option F (copy to file or sysout archiving) is specified. For example:

SYSOUT OP F PRM GPL.%%JOBNAME.D%%ODATE.%%JOBID.T%%TIME

■ SHOUT, DO SHOUT, and DO MAIL

System AutoEdit variables can be used in shouted messages. For example:

MSG JCL ERROR IN JOB %%JOBID %%STEP

■ MEMLIB and OVERLIB

System AutoEdit variables and variables defined in SET VAR parameters can be used in the MEMLIB and OVERLIB fields to specify the appropriate library. Examples of this usage are shown on the following pages.

■ IN, OUT, and DO COND

You can use an AutoEdit variable in a condition name, provided that the AutoEdit variable has a value that is known before the job is ordered.

# **Examples**

### **Date Variables**

#### **Table 248 Date Variables**

Original Scheduling	Current Working	Computer	Format
%%ODATE	%%RDATE	%%DATE	yymmdd
%%ODAY	%%RDAY	%%DAY	dd
%%OMONTH	%%RMONTH	%%MONTH	mm
%%OYEAR	%%RYEAR	%%YEAR	уу
%%OWDAY	%%RWDAY	%%WDAY	n (0-6)
%%OJULDAY	%%RJULDAY	%%JULDAY	jiji

### The original JCL:

```
//PDPA0001 JOB (.....),BILL,CLASS=A
//STEP01 EXEC PDUPDATE
//SYSIN DD *
%%DATE
//
```

#### The JCL submitted on June 6, 2001:

```
//PDPA0001 JOB (.....),BILL,CLASS=A
//STEP01 EXEC PDUPDATE
//SYSIN DD *
010606
//
```

### **ODATE, RDATE and DATE Usage**

```
//PDPA0001 JOB (.....),BILL,CLASS=A
//STEP01 EXEC PDUPDATE
//SYSIN DD *
010606
//
```

#### The original JCL:

```
//PDPA0001 JOB (.....),BILL,CLASS=A
....
//STEP02 EXEC PDPRINT,BUSDATE=%%ODATE
//SYSIN DD *
```

```
EXAMPLE-RDATE=%%RDATE
EXAMPLE-DATE=%%DATE
//
```

On July 24th, we need to run the 22nd, 23rd, and 24th (of the same job) because of delays. On the Active Jobs file we can find three job orders:

```
PDPA0001 of 010722
PDPA0001 of 010723
PDPA0001 of 010724
```

The job of the 22nd is submitted on July 24th at 2300. The result is:

```
//PDPA0001 JOB (.....),BILL,CLASS=A
....
//STEP02 EXEC PDPRINT,BUSDATE=010722
//SYSIN DD *
EXAMPLE-RDATE=010724
EXAMPLE-DATE=010724
//
```

The job of the 23rd is submitted on July 25th at 0025. The result is:

```
//PDPA0001 JOB (.....),BILL,CLASS=A
....
//STEP02 EXEC PDPRINT,BUSDATE=000723
//SYSIN DD *
EXAMPLE-RDATE=010724
EXAMPLE-DATE=010725
//
```

The job of the 24th is submitted on July 25th, 2001 at 0300. The result is:

```
//PDPA0001 JOB (.....),BILL,CLASS=A
....
//STEP02 EXEC PDPRINT,BUSDATE=000724
//SYSIN DD *
EXAMPLE-RDATE=010724
EXAMPLE-DATE=010725
//
```

### **How to Obtain Date Formats – 1**

Format ddmmyy:

```
%%ODAY.%%OMONTH.%%OYEAR
```

Let's follow the variable substitution by stages for June 24, 2001:

```
%%ODAY.%%OMONTH.01
%%ODAY.06.01
```

```
%%0DAY.0601
24.0601
240601
```

Remember Variable substitution is performed from right to left.

A period (.) between two AutoEdit variables is omitted.

### **How to Obtain Date Formats – 2**

Format dd mmm yy (where mmm is the month in character format):

The original JCL:

```
//PDPA0001 JOB (.....),BILL,CLASS=A
//* %%GLOBAL CHARMON
//STEP02 EXEC PDPRT3
//SYSIN DD *
%%ODAY %%MONTH_IN_CHAR_%%OMONTH %%OYEAR
//
```

The CHARMON member (in the CONTROL-M Global library) contains:

```
* MONTHS IN CHAR FORMAT

* W%MONTH_IN_CHAR_01=JAN

%%MONTH_IN_CHAR_02=FEB

%%MONTH_IN_CHAR_03=MAR

.

%%MONTH_IN_CHAR_12=DEC
```

The symbols substitution by stages:

```
%%0DAY %%MONTH_IN_CHAR_%%0MONTH 00
%%0DAY %%MONTH_IN_CHAR_06 00
%%0DAY JUN 00
24 JUN 00
```

The submitted member:

```
//PDPA0001 JOB (.....),BILL,CLASS=A
//* %%GLOBAL CHARMON
//STEPO2 EXEC PDPRT3
//SYSIN DD *
24 JUN 00
//
```

### **How to Obtain Date Formats – 3**

Format *ddmmmyy* (where *mmm* is the month in character format):

According to the preceding example, we might try the following original JCL:

```
//PDPA0001 JOB (.....),BILL,CLASS=A
//* %%GLOBAL CHARMON
//STEP02 EXEC PDPRT3
//SYSIN DD *
%%ODAY.%%MONTH_IN_CHAR_%%OMONTH.%%OYEAR
//
```

Variable substitution by stages would proceed as follows:

```
%%ODAY %%MONTH_IN_CHAR_%%OMONTH.00
%%ODAY %%MONTH_IN_CHAR_06.00
%%ODAY %%MONTH_IN_CHAR_0600
```

However, this results in the following error: Symbol %%MONTH_IN_CHAR_0600 is not resolved.

This error would not have occurred had we tried the following original JCL:

```
//PDPA0001 JOB (.....),BILL,CLASS=A
//* %%GLOBAL CHARMON
//* %%SET %%A=%%MONTH_IN_CHAR_%%OMONTH
//STEP02 EXEC PDPRT3
//SYSIN DD *
%%ODAY.%%A.%%OYEAR
//
```

### How to Obtain the Previous Month's Last Business Date

```
//PDPA0001 JOB (.....),BILL,CLASS=A
//* %%LIBSYM CTM.LIB.SYMBOLS %%MEMSYM LBUSMON
//STEP02 EXEC PDPRT3
//SYSIN DD *
%%LAST_BUSINESS_DATE_IN_PREV_MONTH_OF_%%OMONTH.%%OYEAR
//
```

#### The LBUSMON member in the CTM.LIB.SYMBOLS library contains:

### Variable substitution by stages (during June 2001):

```
%%LAST_BUSINESS_DATE_IN_PREV_MONTH_OF_%%OMONTH.01
%%LAST_BUSINESS_DATE_IN_PREV_MONTH_OF_06.01
%%LAST_BUSINESS_DATE_IN_PREV_MONTH_OF_0601
010531
```

#### – NOTE



An alternate method, which avoids the need to use the MEMSYM member, requires the use of the %%\$WCALC function with the standard working day calendar. For details, see "%%\$WCALC" on page 784.

### **Automatic Job Order for the Next Day**

In many data centers it is necessary to run certain jobs "ahead of time" on a regular basis (such as run today with the business date of tomorrow). The %%\$CALCDTE and %%SUBSTR functions can be used to permit automatic scheduling of such jobs on a daily basis by the CONTROL-M monitor. (The output is in mmddyy format.)

```
//TOMDAILY JOB (....),BILL,CLASS=A
//* %%SET %%A=%%$CALCDTE %%$ODATE +1
//* %%SET %%DD=%%SUBSTR %%A 7 2
//* %%SET %%MM=%%SUBSTR %%A 5 2
//* %%SET %%YY=%%SUBSTR %%A 3 2
//STEP01 EXEC PGM=IKJEFT01,REGION=1000K,DYNAMNBR=30
//SYSPROC DD DISP=SHR,DSN=CONTROL-M-CLIST-LIBRARY
//SYSPRINT DD SYSOUT=*
//SYSTSPRT DD SYSOUT=*
//SYSTSIN DD *
CTMCJOBS SCHEDLIB(CTM.LIB.SCHEDULE) TABLE(SDP00) -
ODAT(%%MM.%%DD.%%YY)
//
```

The %%\$CALCDTE and %%SUBSTR AutoEdit functions can be used for any date calculation that is needed in a production environment.

If you want to use the WAIT FOR ODATE option, which is described in "WAIT FOR ODATE" on page 153, you can use the WAITODAT(YES) parameter.

#### For example

```
CTMCJOBS SCHEDLIB(CTM.LIB.SCHEDULE) TABLE(SDPOO) - ODAT(%%M.%%D.%%Y) WAITODAT(YES)
```

causes the job to wait for a specific date before being processed.

### **Tape Clearance System – Stage 1**

#### The original JCL:

```
//PDPA0001 JOB (.....),BILL,CLASS=A
//* %%LIBSYM CTM.LIB.SYMBOLS %%MEMSYM TAPES
//STEP02 EXEC PDINP3
//S001.INPUT DD VOL=SER=%%FEDERAL_BANK_TAPE
//
```

#### The member TAPES in the CTM.LIB.SYMBOLS library contains:

```
* EXTERNAL TAPES LIST

*
%%FEDERAL_BANK_TAPE=045673

%%IRS_TAPE=XXXXX

%%STOCK_EXCHANGE_TAPE=YYYYYY

.
.
```

#### The submitted JCL:

```
//PDPA0001 JOB (.....),BILL,CLASS=A
//* %%LIBSYM CTM.LIB.SYMBOLS %%MEMSYM TAPES
//STEP02 EXEC PDINP3
//S001.INPUT DD VOL=SER=045673
//
```

The use of a central member for all external tapes is a very simple management tool. The minute a tape arrives, its number is typed in the member, and the tape is sent to the computer room. There is no need to keep the tapes "at hand" on the schedulers' table until the job is submitted. The function of receiving tapes can be centralized, controlled, and independent of the production process.

### **Tape Clearance System – Stage 2**

The example provided on the previous page has one basic weakness. It cannot handle delays. If a certain job does not run one day, and on the next day it must be run twice (once for each execution day), there is a danger of overriding the tape number in the control member. To solve this problem, let's improve our tape clearance system.

```
//PDPA0001 JOB (.....),BILL,CLASS=A
//* %%LIBSYM CTM.LIB.SYMBOLS %%MEMSYM TAPE%%OMONTH.%%ODAY
//STEP02 EXEC PDINP3
//S001.INPUT DD VOL=SER=%%FEDERAL_BANK_TAPE
//
```

The TAPE0714 member in the CTM.LIB.SYMBOLS library contains:

```
:
*
* EXTERNAL TAPES LIST

*
%%FEDERAL_BANK_TAPE=045673

%%IRS_TAPE=XXXXX

%%STOCK_EXCHANGE_TAPE=YYYYYY

.
```

There are other advantages:

- The ability to roll back several dates without losing the dynamic parameters.
- Complete documentation of a tape's usage.

Use a CONTROL-M job to automatically create a member, TAPE*mmdd*, for each scheduling date, based on a master copy. For example:

```
// EXEC PGM=IEBCOPY
.
//SYSIN DD *
C I=IN, O=OUT
S M=((TAPES,TAPE%%OMONTH.%%ODAY))
```

### **Tape Management System**

The original JCL:

```
//PDPA0001 JOB (.....),BILL,CLASS=A
//* %%LIBSYM CTM.LIB.TAPES %%MEMSYM PDTAPES
//STEP02 EXEC PDBKP3
//S001.OUTPUT DD VOL=SER=%%PDTAPE_0001_%%OWDAY
//
```

### The PDTAPES member in the CTM.LIB.TAPES library contains:

```
* TAPES OF DP APPLICATION

* 
%%PDTAPE_0001_1=045673

%%PDTAPE_0001_2=045683

%%PDTAPE_0001_3=045677

%%PDTAPE_0001_4=043433

%%PDTAPE_0001_5=045543

%%PDTAPE_0001_6=045556

%%PDTAPE_0001_7=045666

* 
%%PDTAPE_0010_1=046600
```

We have created a cycle of seven tapes to be used by this application.

### **Dynamic Job Name**

#### The required format:

```
//PDPAddxx JOB (.....),BILL.CLASS=A
```

where *dd* is the business day of the month, and *xx* varies according to the job in the application.

#### The solution:

```
//PDPA%%ODAY%%.xx JOB (....),BILL,CLASS=A
```

### **Controlling the Target Computer by Class**

CONTROL-M can dynamically decide to which computer to send a job. The following examples demonstrate the relation between CONTROL-M resource acquisition parameters and local JCL standards implementation.

#### – NOTE –



This example assumes that a \$ generic resource was specified in the job scheduling definition. For more information, see "Resource Allocation in Multi-CPU Environments" on page 598

```
//* %%GLOBAL CLASSES
//PDPA0001J0B(....),BILL,CLASS=%%FAST_CLASS_OF_%%$SIGN
```

### The CLASSES member in the CONTROL-M Global library contains:

```
* DEFINITIONS OF CLASSES IN THE COMPUTERS

* 
%%FAST_CLASS_OF_1=A

%%FAST_CLASS_OF_2=D

%%FAST_CLASS_OF_3=K

%%SLOW_CLASS_OF_1=W

.
```

### Controlling the Target Computer by System Affinity



#### - NOTE

This example assumes that a \$ generic resource was specified in the job scheduling definition. For more information, see "Resource Allocation in Multi-CPU Environments" on page 598

```
//* %%GLOBAL SYSAFF
//PDPA0001 JOB (.....),BILL,CLASS=A
/*J SYSAFF=%%NAME_OF_COMPUTER_%%$SIGN
```

### The SYSAFF member in the CONTROL-M Global library contains:

```
* NAMES OF THE COMPUTERS

* %%NAME_OF_COMPUTER_1=SYSA

%%NAME_OF_COMPUTER_2=SYSB

%%NAME_OF_COMPUTER_3=TEST
```

#### The submitted JCL (for CPU ID 2):

```
//* %%GLOBAL SYSAFF
//PDPA0001 JOB (.....),BILL,CLASS=A
/*J SYSAFF=SYSB
```

### **%%BLANKn Statement**

A program expects to receive the day of the week and the time of day as structured input:

- Day of the week in column 1
- Time of day in column 11

#### The original JCL:

```
//PDPA0001 JOB (.....),BILL,CLASS=A
//* %%LIBSYM CTM.LIB.SYMBOLS %%MEMSYM DAYTIME
....
//STEP02 EXEC PDPRT3
//SYSIN DD *
%%H%%OWDAY%%.%%TIME
//
```

The DAYTIME member in the CTM.LIB.SYMBOLS library contains the following AutoEdit user symbols:

```
%%H1=SUNDAY%%BLANK4
%%H2=MONDAY%%BLANK3
%%H4=WEDNESDAY%%BLANK1
%%H5=THURSDAY%%BLANK1
%%H5=THURSDAY%%BLANK4
%%H0=SATURDAY%%BLANK2
Variable substitution by stages:
%%H4%0WDAY%%.%%TIME
%%H%%0WDAY%.%%TIME
%%H%%0WDAY .085300
SUNDAY 085300
The submitted JCL:
//PDPA0001 JOB (.....).BILL,CLASS=A
//* %%LIBSYM CTM.LIB.SYMBOLS %%MEMSYM DAYTIME
...
//STEP02 EXEC PDPRT3
//SYSIN DD *
SUNDAY 085300
//
```

### **%%RANGE Statement**

#### The original JCL:

```
//PDPA0001 JOB (.....),BILL,CLASS=A
//* %%LIBSYM CTM.LIB.SYMBOLS %%MEMSYM LBUSMON
//STEP02 EXEC PDPRT3
//* + + + +
//SYSIN DD *
%%LAST_BUSINESS_DATE_IN_%%OMONTH REPORT1
//
```

The constant REPORT must be in column 40.

#### The submitted JCL:

```
//PDPA0001 JOB (.....),BILL,CLASS=A
//* %%LIBSYM CTM.LIB.SYMBOLS %%MEMSYM LBUSMON
//STEP02 EXEC PDPRT3
//* + + + +
//SYSIN DD *
030400 REPORT1
//
```

#### The correct solution:

```
//PDPA0001 JOB (.....),BILL,CLASS=A
//* %%LIBSYM CTM.LIB.SYMBOLS %%MEMSYM LBUSMON
//STEP02 EXEC PDPRT3
//* + + + +
//* %%RANGE 1 39
//SYSIN DD *
%%LAST_BUSINESS_DATE_IN_%%OMONTH REPORT1
//
```

### **SYSIN Parameter Containing %%**

### Disabling AutoEdit Resolution

#### The original JCL:

```
//PDPA0001 JOB (.....),BILL,CLASS=A

//STEP02 EXEC PDPRT3

//SYSIN DD *
%%VAR Do not resolve the AutoEdit variable on this line.

// EXEC ... PARM='%%ODATE'

//
```

#### The solution:

```
//PDPA0001 JOB (.....),BILL,CLASS=A
//STEP02 EXEC PDPRT3
//SYSIN DD *
%%RESOLVE OFF
%%VAR Do not resolve the AutoEdit variable on this line.
%%RESOLVE YES
// EXEC ... PARM='%%ODATE'
//
```

If %%RESOLVE=NO is specified, the line is submitted as is.

### %%INCLIB and %%INCMEM Statements

The original JCL:

```
//PDPA0001 JOB (.....),BILL,CLASS=A
//STEP01 EXEC PDPRPT1
....
//* %%INCLIB CTM.LIB.COMJCL %%INCMEM PDPRPT2
//
```

The PDPRPT2 member in the CTM.LIB.COMJCL library contains:

```
//STEP02 EXEC PDPRPT2
//SYSIN DD *
%%DATE
```

The submitted JCL (on June 6, 2001):

```
//PDPA0001 JOB (.....),BILL,CLASS=A
//STEP01 EXEC PDPRPT1
....
//* %%INCLIB CTM.LIB.COMJCL %%INCMEM PDPRPT2
//STEP02 EXEC PDPRPT2
//SYSIN DD *
010606
//
```

### Boolean "IF" Logic

### **Example 1**

This example illustrates CONTROL-M's ability to perform Boolean "IF" logic.

The original JCL:

```
//PDPA0001 JOB (.....),BILL,CLASS=A
//*
//* %%IF %%TIME LT 120000
//* %%SET %%PGMA=MORNPGM
//* %%ELSE
//* %%SET %%PGMA=AFTPGM
//* %%ENDIF
//*
//STEP01 EXEC PGM=%%PGMA
...
//
```

I

The submitted JCL at 1:00 p.m:

```
//PDPA0001 JOB (.....),BILL,CLASS=A

//
//* %%IF %%TIME LT 120000

//* %%ELSE
//* %%SET %%PGMA=AFTPGM
//* %%ENDIF
//
//STEP01 EXEC PGM=AFTPGM
...
//
```

The %%IF expression is not true since it is past 12:00 noon; therefore, the statements following %%ELSE are executed. The program executed in STEP01 is AFTPGM.

### **Example 2**

This example illustrates the use of CONTROL-M's conditional logic in conjunction with CONTROL-M "INCLUDE" and "GO TO" logic.

```
//PDPA0001 JOB (.....),BILL,CLASS=A
//*
//* %%IF %%WDAY NE 1
//* %%GOTO RUN_DAILY
//* %%ELSE
//* %%INCLIB CTM.LIB.COMJCL %%INCMEM MONTHLY
//* %%ENDIF
//*
//* %%LABEL RUN_DAILY
//* %%LABEL RUN_DAILY
...
//
```

The MONTHLY member in the CTM.LIB.COMJCL library contains:

```
//STEPMON EXEC PGM=MONTHLY
...
```

On the first day of the month both the DAILY and MONTHLY programs run. The submitted JCL:

```
//PDPA0001 JOB (.....),BILL,CLASS=A
//*
//* %%IF 1 NE 1
//* %%ELSE
//* %%INCLIB CTM.LIB.COMJCL %%INCMEM MONTHLY
//STEPMON EXEC PGM=MONTHLY
...
//* %%ENDIF
//*
//* %%LABEL RUN_DAILY
//STEPDAI EXEC PGM=DAILY
...
//
```

### On any other day of the month, only the DAILY program runs. The submitted JCL:

```
//PDPA0001 JOB (.....),BILL,CLASS=A
//*
//* %%IF 2 NE 1
//* %% GOTO RUN_DAILY
//* %%ELSE
//* %%ENDIF
//*
//* %%LABEL RUN_DAILY
//* %%LABEL RUN_DAILY
//* %%LABEL RUN_DAILY
//STEPDAI EXEC PGM=DAILY
...
//
```





# **Selected Implementation Issues**

This chapter includes the following topics:			
Overview	12		
Job Ordering Methods 8	12		
Job Ordering Through Quick Submit Command CTMQSB 8			
Special Purpose Job Ordering From Special Environments: CTMAJO 8	15		
Manual Conditions File and Maybe Jobs 8	17		
Loading the Manual Conditions File 8	17		
Using the Manual Conditions File 8	17		
Handling Manual Conditions			
Handling Unscheduled Conditions 8			
Handling Maybe Dependencies	19		
Maybe Jobs in Group Scheduling Tables 8	20		
MAINVIEW Batch Optimizer Considerations	21		
Job-Related Considerations for Pipes 8	21		
Enhanced Runtime Scheduling Algorithm			
System-Related Considerations 8	23		
Parameter Prompting Facilities			
Parameter Prompting Facility – Type 1			
Summary 8			
Usage Notes			
Parameter Prompting Facility—Type 2	29		
Maintenance Utilities			

### **Overview**

This chapter provides you with concepts and hints for successful implementation of CONTROL-M. It also provides a detailed description of the procedures required for implementation of CONTROL-M by the user and operator. The following implementation concepts and instructions are discussed in this chapter:

- alternative methods of job ordering
- Manual Conditions list and Maybe jobs
- MAINVIEW Batch Optimizer (MVBO) considerations
- parameter prompting facilities

For information about the INCONTROL administrator's implementation of CONTROL-M, see the *INCONTROL for z/OS Administrator Guide*.

#### - NOTE -



Topics in this chapter require familiarity with background information presented in Chapter 1, "Introduction to CONTROL-M," and familiarity with relevant information presented in other chapters.

# **Job Ordering Methods**

Under CONTROL-M, job ordering is normally performed automatically by the New Day procedure and User Daily jobs during New Day processing, as described in detail in the *INCONTROL for z/OS Administrator Guide*.

However, at times it is desirable to perform job ordering using methods other than the New Day procedure and User Daily jobs. For example, it may be necessary to schedule special purpose jobs, or to order jobs for a different working date.

CONTROL-M provides several alternative methods of job ordering. Some methods of job ordering are performed online; others in batch. Some are performed automatically, while others are performed manually.

Below is a list of facilities and functions that enable jobs to be ordered without the New Day procedure and User Daily jobs.

**Table 249 Alternative Job Ordering Methods** 

Method	Description	
Table/Job List screen	Enables jobs to be ordered from Online facility screens. For more information, see "Ordering (Scheduling) Jobs" on page 150	
Job Order panel	Allows job ordering through the online utility (or CLIST) CTMJOBRQ. For more information, see "M1: Issue a Job Order" on page 326	
End User Job Order interface	Allows job ordering through the online utility (or CLIST) CTMJBINT. For more information, see "M6: End-User Job Order Interface" on page 370	
Utility CTMJOB	Described in the INCONTROL for z/OS Utilities Guide.	
Utility CTMBLT	Described in the INCONTROL for z/OS Utilities Guide.	
TSO CLIST	Allows job ordering directly from the TSO environment. For more information, see the description of the CTMJOB utility in the CONTROL M chapter in the <i>INCONTROL for z/OS Utilities Guide</i> , which includes an example of such job ordering.	
Quick submit command CTMQSB	Allows job ordering through CONTROL-M submit command CTMQSB (instead the ISPF submit command). For more information, see "Job Ordering Through Quick Submit Command CTMQSB" on page 813.	
Job ordering from special environments	Facilitates job ordering from other environments (CICS, ROSCOE, and so on). For more information, see "Special Purpose Job Ordering From Special Environments: CTMAJO" on page 815.	

### Job Ordering Through Quick Submit Command CTMQSB

In many instances, the contents of the job are determined by an end user before submission. For example, a user may maintain a member that contains the JCL and parameters of a certain report. When someone requests the report, the user edits the member, possibly using ISPF, changes the parameters of the report, and uses the ISPF SUBMIT command to submit the job.

As described in the previous paragraphs, CONTROL-M can detect such jobs when they appear on spool, and control their execution. However, there are a few disadvantages to this method. The primary disadvantage is in handling job abends. When an On Spool job abends, it is not clear which JCL member must be submitted to perform a rerun. For example, in the above example, if the JCL has not been saved, such as where the user exited ISPF EDIT using the CANCEL command, there is no original member from which to perform the rerun.

This problem can be overcome using CONTROL-M command CTMQSB. When submitting a job, use command CTMQSB instead of the regular ISPF SUBMIT command. Just type it in the command line and press **Enter**. You may have to prefix it by the % character to designate a CLIST.

It is possible to replace the ISPF SUBMIT command with the CONTROL-M CTMQSB command. For more information, see the description of installing ISPF support in the CONTROL-M Installation Procedure in the *INCONTROL for z/OS Installation Guide*.

If the member contains JCL cards that start with //*CONTROLM then special processing takes place, that is, the member is not submitted. The CONTROL-M submit command looks for two //*CONTROLM cards with the following format (order and position in the job are not important):

//*CONTROLM TABLE scheduling-tables-library table-name //*CONTROLM JCL JCL-library

The current JCL member is written to the specified JCL library. The name of the member is composed of the first three letters of the TSO user ID, and the CONTROL-M order ID (5 characters). This method ensures that the name is unique.

The scheduling table is read from the specified library. The submit command assumes that the table contains only one job scheduling definition. If the table contains definitions for more than one job, only the first job scheduling definition is taken into account; the remainder are ignored. The CTMQSB CONTROL-M command replaces the original library and member names with the names of the JCL library and member where the job has been stored, as described in the preceding section. If the WM1822 optional wish is applied, the user ID (OWNER) of the job is replaced by the TSO user ID. The WM1822 optional wish is in the IOADFLT member in the IOA IOAENV library.

To avoid accumulation of old members, it is advisable that a new, empty JCL library be used each day.

CONTROL-M job order security exit CTMX001 is invoked (as under CTMJOBRQ). If the job order is valid, it is placed on the CONTROL-M Active Jobs file. The job is then submitted based on the regular job scheduling criteria, such as IN, CONTROL, TIME.

Scheduling tables that are referred to by //*CONTROLM statements must not be included in a batch User Daily or in New Day processing. They must contain a skeleton of a job order, such as reports that require IMS to be up, reports that use substantial IDMS resources, update to certain VSAM files, and so on.

It is possible to force the use of the CONTROL-M submit facility. When the CONTROL-M CTMQSB command is activated, the contents of the member to be submitted are passed to CONTROL-M User Exit CTMX010. This exit can automatically add //*CONTROLM cards to the submitted member. Use of this technique results in a completely scheduled environment. All submitted jobs are under CONTROL-M control.

#### - NOTE



Each member processed using the command CTMQSB must contain only one job. If one of these members contains more than one job, all the jobs are submitted; however, a message is produced for only the last job. If the job is ordered, CONTROL-M submits all the jobs in the member, but controls only the first job.

CONTROL-D users: The D-CAT field of the job scheduling table is ignored for jobs that are scheduled using the CONTROL-M CTMQSB command. This means that a report decollating mission is not automatically ordered for jobs that are scheduled by the CTMQSB command.

If CTMQSB is being used to order jobs and not simply for quick submission, then the AJF must be allocated (via DDname DACKPT) in the TSO environment or from within the CTMSETSB Clist. For more information, see the section about installing ISPF support in the *INCONTROL for z/OS Installation Guide*.

# Special Purpose Job Ordering From Special Environments: CTMAJO

This section describes a special program, CTMAJO, that is supplied with CONTROL-M. CTMAJO was designed to handle a situation that sometimes arises, when the user needs to order special jobs from any of various environments, such as CICS or ROSCOE.

#### WARNING



CTMAJO will not be supported in future versions of CONTROL-M. BMC recommends that you already use the CTMBLT utility (which provides improved functionality) instead of CTMAJO. You use the CTMBLT utility to dynamically build job scheduling definitions and to order individual jobs when required. For more information, see the *INCONTROL for z/OS Utilities Guide*.

Since program CTMAJO is not environment-dependent, the INCONTROL administrator must develop an application that enables the program to be used with the particular environment. One such application, CTMQSB, is supplied with CONTROL-M. CTMQSB is for use with CTMAJO under ISPF, and is described in "Job Ordering Through Quick Submit Command CTMQSB" on page 813.

The CTMAJO program accepts the following parameters:

- the JCL of the job, which is already loaded into memory
- the name of a special purpose JCL library in which to place the JCL
- the name of a scheduling library
- the name of a scheduling table (in the above library) containing a single, skeletal, job scheduling definition
- the requested scheduling date

To handle the special purpose request, CTMAJO performs the following:

- takes the JCL of the job to be submitted from memory and writes it to the specified single purpose JCL library, using a unique member name
- takes the skeletal job scheduling definition from the scheduling table in the scheduling library, and loads the job scheduling definition to the Active Jobs file

When placing the job order in the job scheduling definition, CONTROL-M overwrites the MEMNAME value from the skeleton with the name of the special purpose JCL member. It also specifies the requested scheduling date.

The job then comes under the control of the CONTROL-M monitor:

- If runtime scheduling criteria are specified in the skeletal job scheduling definition, the job is not submitted until those criteria are satisfied.
- If post-processing parameters are specified, they are performed upon completion of the job.

Using CTMAJO to order special purpose jobs under special environments is preferable to bringing jobs under CONTROL-M control as On Spool jobs because when CTMAJO is used, the JCL is available, if necessary, for job rerun. With On Spool jobs, the JCL member may not be known.

For a sample call to the CTMAJO utility, see the ROSORDER member in the IOA SAMPLE library.

## **Manual Conditions File and Maybe Jobs**

The Manual Conditions file contains a list of prerequisite conditions that are required by jobs in the Active Jobs file but which are not available, that is, added to the IOA Conditions file, unless there is some form of user intervention.

### **Loading the Manual Conditions File**

Conditions are added to the Manual Conditions file through the IOALDNRS utility. This utility is run during New Day processing, but it must also be run following the addition of a set of jobs in the Active Jobs file.

The IOALDNRS utility checks the IN conditions required by scheduled jobs against the conditions available in the IOA Conditions file and against the OUT conditions that can be set by the scheduled jobs.

All IN conditions that are not in the IOA Conditions file and that are not listed as OUT conditions in a scheduled job are added to the Manual Conditions file.

### **Using the Manual Conditions File**

The Manual Conditions file provides the user with a list of conditions for which manual intervention is required if the conditions are to be added to the IOA Conditions file.

To utilize this list effectively, the user must distinguish between two types of conditions in the list because each requires a different type of intervention. From the user perspective, the two types of conditions are:

#### Manual Conditions

Conditions that always require manual intervention and are therefore never automatically added by jobs as OUT or DO COND conditions.

#### **Example**

Job-X, which requires that a tape has arrived before the job is submitted, contains IN prerequisite condition TAPE-ARRIVED.

This condition must not be automatically added to the IOA Conditions file by a job, but must instead be manually added by the operator only after the tape has arrived.

#### Unscheduled Conditions

Conditions that can be added automatically by a job, but which appear in the Manual Conditions list because none of the jobs scheduled that day set the condition.

#### Example

Job-B requires IN condition JOB-A-ENDED-OK. This condition is added as an OUT condition by Job-A. Job-B is scheduled on a day during which Job-A is not scheduled.

The distinction between the two types of conditions mentioned above is important because each type requires a different user response, as described below.

### **Handling Manual Conditions**

The handling of Manual Conditions, as defined above, is fairly straightforward. In the above example, the user clearly does not want the condition added automatically, nor does the user want the condition ignored. Simply put, Job-X must not be run unless the required tape has arrived at the site, in which case the operator adds the condition. For this type of condition, the only desired manual intervention is the adding of the condition at the appropriate time. This can be performed by option A (Add) in the Manual Conditions screen.

### **Handling Unscheduled Conditions**

The handling of Unscheduled Conditions, as defined above, is more complex because it concerns the issue of normal dependency versus "Maybe" dependency:

#### Normal Dependency

A successor job is always dependent on the predecessor job, regardless of whether the predecessor job is scheduled.

With this type of dependency, using the example cited above, successor Job-B must not be submitted because predecessor Job-A was not scheduled and executed.

In this case, the dependency must not be ignored. The unscheduled prerequisite condition is not added manually.

#### ■ "Maybe" Dependency

A successor job is dependent on the predecessor job only if the predecessor job is scheduled that day. If the predecessor job is not scheduled that day, the successor job can still be submitted, provided that other runtime scheduling criteria are satisfied.

In this case, the predecessor job is referred to as a Maybe job.

With this type of dependency, using the example cited above, successor job Job-B must be submitted, provided all other runtime scheduling criteria are satisfied, because predecessor job Job-A was not scheduled.

In this case, the dependency must be ignored or bypassed. Methods for ignoring Maybe dependencies are described below.

### **Handling Maybe Dependencies**

The most common method of handling Maybe job dependencies is to add the unscheduled conditions of Maybe jobs to the IOA Conditions file.

However, examining each condition in the Manual Conditions list to determine if it is an unscheduled condition from a Maybe job, and manually adding each Maybe job unscheduled condition, is a difficult process. The process can be greatly simplified and automated, by following these steps:

1. Define a Unique Prefix for Maybe Job Prerequisite Conditions

When Maybe dependencies are defined, the prerequisite IN, OUT and DO COND conditions must all have the same unique prefix (that is, a prefix that is not used for other prerequisite conditions).

Using a unique prefix symbol makes it easier to see unscheduled conditions of Maybe Jobs in the Manual Conditions list.

Normally, this prefix is either symbol # or @.

#### NOTE -



If your site utilizes MVS restarts and uses symbol @ in OUT conditions for the restart, this symbol must not be used as a prefix for Maybe job conditions. In this case, use the # symbol for Maybe conditions. For details, see Appendix E, "MVS Job Restart Without CONTROL-M/Restart."

2. Use the ADDMNCND KeyStroke Language utility to add the prerequisite conditions. The ADDMNCND KSL utility automatically adds all conditions with a specified prefix in the Manual Conditions file to the IOA Conditions file.

By specifying the above-defined unique prefix symbol in the utility, unscheduled conditions from Maybe jobs are automatically added, making manual adding of the conditions unnecessary.

After the above two steps have been implemented, the only manual intervention required for unscheduled conditions of Maybe jobs is the executing of the ADDMNCND KSL utility.

### **Maybe Jobs in Group Scheduling Tables**

The above implementation for handling unscheduled conditions of Maybe jobs can be applied to jobs and conditions in all types of scheduling tables.

However, an alternative method is available for conditions and jobs in Group scheduling tables. Rather than add the unscheduled conditions of Maybe jobs to the IOA Conditions file, the unscheduled conditions can be removed as runtime scheduling criteria for the successor job orders.

The Group Entity definition in Group scheduling tables contains an ADJUST CONDITIONS field. If a value of Y is specified in the ADJUST CONDITIONS field, CONTROL-M checks the scheduled jobs for unscheduled conditions.

Unscheduled conditions normally added by other jobs in the same Group scheduling table are removed from the IN statements of the scheduled job orders:

- These conditions do not appear in the Zoom screen. They are not, however, deleted from the original job scheduling definition.
- These conditions also do not appear in the Manual Conditions file. Therefore, there is no real advantage to defining them with a unique prefix, unless they are used as IN conditions for jobs in a different table.

Note the following points

- Unscheduled conditions normally added by jobs in other scheduling tables are not removed from the job order. They appear in the Manual Conditions file.
- As indicated above, ADJUST CONDITIONS applies only to jobs in the same Group scheduling table. By contrast, the IOALDNRS utility detects unscheduled conditions of Maybe jobs across scheduling tables, and the ADDMNCND KSL utility adds these conditions to the IOA Conditions file regardless of scheduling table.

For more information, see Chapter 3, "Job Production Parameters."

### **MAINVIEW Batch Optimizer Considerations**

MAINVIEW Batch Optimizer (MVBO) is a batch optimization system that enables effective parallel processing and efficient resource utilization in the mainframe environment. The Job Optimizer Pipes component of MVBO enhances this capability using MVS Pipe technology. If MVBO/Job Optimizer Pipes is installed at your site, you can include the CONTROL-M PIPE parameter in a job scheduling definition in order to use MVBO/Job Optimizer Pipes functionality.

### **Job-Related Considerations for Pipes**

The following job-related issues must be considered when using the PIPE parameter in a CONTROL-M job scheduling definition:

- When using pipes for jobs submitted by CONTROL-M, the PIPE parameter must be used if parallel submission of all pipe participants is to be ensured.
- Normally (that is, when pipes are not used), prerequisite conditions ensure desired flow of predecessor and successor jobs.
  - However, when values are specified in the PIPE parameters of interrelated job scheduling definitions, CONTROL-M uses them to create Collections. The jobs in a Collection are not submitted until all prerequisite conditions required by all jobs in the Collection are satisfied. If a dependency exists between an OUT condition of one job and an IN condition of another job in the same Collection, it prevents submission of all the jobs in the Collection. CONTROL-M resolves this problem by ignoring the IN condition, thus bypassing the dependency between the jobs in the Collection and enabling the submission of the jobs. If the job is removed from the Collection, its ignored IN conditions reappear.
- When two jobs in the same Collection request the same Control resource, and at least one of them requests the resource in "exclusive" mode, a "deadlock" situation arises—the Collection jobs are not submitted. To prevent this, CONTROL-M ignores the Control resource requests of one of the jobs, as follows: If one of the jobs requested the resource in "shared" mode, that resource request is ignored; if both jobs requested the resource in "exclusive" mode, the resource request of the job with the shorter average elapsed time is ignored. If the job is removed from the Collection, its ignored Control resources reappear.

### <u> — NC</u>



To allow integration between CONTROL-M and MVBO/Job Optimizer Pipes, the pipe name specified in the CONTROL-M job scheduling definition must be identical to the pipe name specified in the MVBO/Job Optimizer Pipes rule. Currently, this requirement is not forced by CONTROL-M.

 Jobs cannot be run in parallel if TIME FROM and TIME UNTIL specifications for the jobs do not overlap.

This case must be considered individually at time of implementation.

- When PIPE definitions are added, the Quantitative resources defined for the jobs in the Collection must be checked to see if some of the defined resources are no longer necessary. For example, if a pipe replaces a tape data set, the tape resource may not be required. Such resources must be removed from the job scheduling definition.
- In a non-Sysplex environment, all jobs that are part of a Collection must run in the same system. Therefore, BMC Software recommends that you avoid using resources prefixed by a dollar sign (\$) in jobs that are part of a Collection, to ensure that all the jobs are submitted to the same system.

### **Enhanced Runtime Scheduling Algorithm**

When jobs that are part of a Collection are scheduled, CONTROL-M treats the Collection as one unit of work for processing runtime scheduling criteria in the following ways:

- CONTROL-M ensures that the required number of participants, as defined for the pipe in the MVBO/Job Optimizer Pipes rule, access each pipe in the Collection. If a participant is missing, the jobs in the Collection are not submitted. This ensures that a job is not submitted when its participants are not scheduled on that day.
- CONTROL-M analyzes all resources, such as prerequisite conditions, Quantitative resources, and time limits, required by all jobs in the Collection as a single set. All participants are submitted together when all the resources required by the set are available. This ensures the parallel submission of all pipe participants.

To ensure that the jobs begin execution on time, it is recommended that initiators be handled as Quantitative resources. This ensures that submitted jobs do not wait for initiators and delay other jobs in the Collection.

■ CONTROL-M checks if the MVBO/Job Optimizer Pipes monitor is active before submitting the jobs in the Collection. If the MVBO/Job Optimizer Pipes monitor is not active, the jobs in the Collection are not submitted. This ensures that, when submitted, the jobs run in parallel, using pipes.

### **System-Related Considerations**

The following system-related issues must be considered when using the PIPE parameter in a CONTROL-M job scheduling definition.

■ Parallel processing changes resource usage in the system. All resources required for all the jobs in the Collection must be available when the jobs are submitted.

This means that more resources, such as initiators, tape drives, CPUs, are required for shorter time periods; they become available after the jobs using the pipe finish execution. Therefore, when using pipes, resource usage in the system in which the jobs are to run must be reviewed to ensure that all required system resources are available at the time the jobs are submitted.

The change in resource usage may necessitate changing the maximum quantities defined for CONTROL-M to satisfy the changed requirements.

## **Parameter Prompting Facilities**

It is assumed that the reader is familiar with the following CONTROL-M facilities and concepts:

- JCL and AutoEdit facility
- prerequisite conditions
- Manual Conditions (Screen 7) and the IOALDNRS utility

CONTROL-M provides two different types of Parameter Prompting facilities. The online use of the two Parameter Prompting facilities is described in Chapter 2, "Online Facilities."

This chapter provides an explanation of how the Parameter Prompting facilities work, how they differ from each other, and how to choose the facility that best suits your operational needs. In addition, certain preparatory steps are detailed.

### **Parameter Prompting Facility – Type 1**

The Parameter Prompting facility – Type 1 is an ISPF table-based facility that provides automatic prompting for AutoEdit parameter values and setting of prerequisite conditions. It is the recommended method for operations personnel to automate the updating of AutoEdit parameter members.

The CONTROL-M JCL and AutoEdit facility eliminates the need for frequent manual changes to job parameters. However, there are usually a few job parameter changes that cannot be automated, for example, tape serial numbers, which are unknown prior to tape arrival. These types of parameters require manual modification by the user, generally operations personnel.

#### **Old Method**

To illustrate how prior versions of CONTROL-M solved this problem, consider the daily arrival of IRS tape number 123456.

JCL for JOBA

JCL for JOBA

JCL for JOBA

JOBA

//

**LIBSYM CTM.LIB.SYMBOLS **MEMSYM TAPES

//

//

**EIRS_TAPE=XXXXXX

**

Member TAPES

in library CTM.LIB.SYMBOLS

Figure 357 Illustration 1A: How CONTROL-M Formerly Handled A New Tape

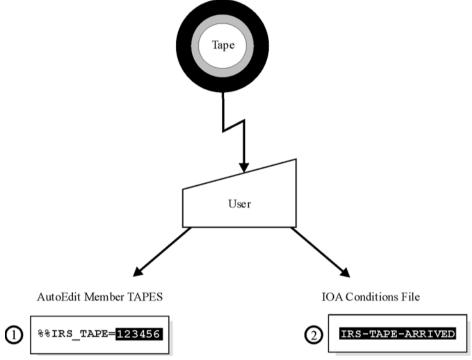
The illustration above represents the one-time definitions required to prepare CONTROL-M for handling the IRS tape.

- JOB A requires input of the IRS tape number before it can run. The job must be defined in a CONTROL-M scheduling table with an IN prerequisite condition of IRS-TAPE-ARRIVED.
- 2. The JCL for JOB A must include %%LIBSYM and %%MEMSYM control statements pointing to the AutoEdit Library CTM.LIB.SYMBOLS and the AutoEdit member TAPES.

3. The AutoEdit member TAPES contains several AutoEdit parameters (from various jobs), including the parameter %%IRS_TAPE.

On a given day, the Manual Conditions file created by the IOALDNRS utility indicates that the prerequisite condition IRS-TAPE-ARRIVED must be added manually by the user. This serves as a reminder to the operations personnel that a job is waiting for an IRS tape number. When the tape arrives, the user must perform two steps, as illustrated in the following figure:

Figure 358 Illustration 1B: Steps Formerly Performed by the User



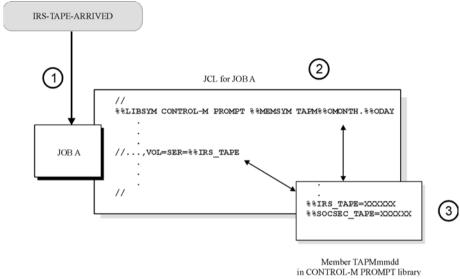
- 1. Access the AutoEdit member TAPES and assign value 123456 to the %%IRS_TAPE parameter.
- 2. Enter Screen 7 to manually add condition IRS-TAPE-ARRIVED.

When the condition IRS-TAPE-ARRIVED has been added to the IOA Conditions file, and assuming all other runtime conditions are met, the CONTROL-M monitor submits the job. When the job is submitted, the value of %%IRS_TAPE in the JCL of JOB A is updated by the value in the TAPES member. The job parameter VOL=SER=%%IRS_TAPE resolves to VOL=SER=123456.

#### **New Method**

In the current version, the same problem is resolved in a different way using the Parameter Prompting facility – Type 1.

Figure 359 Illustration 2A: How CONTROL-M Now Handles A New Tape



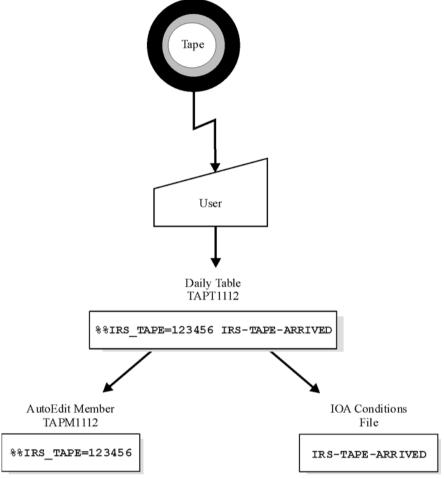
The illustration above represents the one-time definitions required to prepare CONTROL-M for handling the IRS tape when using Parameter Prompting facility – Type 1.

- 1. JOB A requires input of the IRS tape number before it can run. The job must be defined in a CONTROL-M scheduling table with IN prerequisite condition IRS-TAPE-ARRIVED.
- 2. The JCL for JOB A includes %%LIBSYM and %%MEMSYM control statements pointing to the CONTROL-M PROMPT prompting parameters library and the TAPM%%OMONTH.%%ODAY daily AutoEdit member.
- 3. Using the first option of the Parameter Prompting facility Type 1, groups of AutoEdit parameters that require value assignment are defined once. These parameters are grouped into a Master Prompting table, the Master table. Default parameter values may be assigned. In addition, prerequisite conditions to be associated with parameters are designated. In this example, several parameters from various jobs have been defined in the TAP Master table, including the %%IRS_TAPE parameter from JOB A. Prerequisite condition IRS-TAPE-ARRIVED has been associated with this parameter.

When the tape arrives, the user only performs one step (illustrated below):

Figure 360 Illustration 2B: Single Step Now Performed by the User





The user selects the TAP table from a list of Master tables and is presented with Daily Prompting table TAPT1112, an automatically created copy of the Master table for the current date. The Daily Prompting table consists of parameter names, (optional) descriptions, and default values. The user updates the %%IRS_TAPE parameter with the value 123456.

The facility automatically adds condition IRS-TAPE-ARRIVED to the IOA Conditions file and updates the daily AutoEdit member TAPM1112.

### **Summary**

By using Parameter Prompting facility – Type 1, it is only necessary to update the Daily table. The user no longer needs to remember which AutoEdit parameters in which AutoEdit symbol member require changing, nor the prerequisite conditions that require setting. The Parameter Prompting facility automatically handles updating of the AutoEdit member, and adds any required conditions to the IOA Conditions file.

### **Usage Notes**

### **JCL Modifications**

JCL members for jobs containing AutoEdit parameters defined in Master tables must be modified as follows:

- The %%LIBSYM control statement must point to the CONTROL-M PROMPT library.
- The %%MEMSYM control statement member name must be in the following format:

```
@@@M%%OMONTH.%%ODAY
```

where @@@ is the Table Name Prefix defined in Option 1 of the facility. %%OMONTH.%%ODAY can be replaced with any date variable or date constant in the format mmdd.

### The IOALDNRS Utility

The Parameter Prompting facility automatically adds conditions to the IOA Conditions file after parameter update. These conditions no longer require submission through Screen 7, and therefore do not need to appear on the Manual Conditions file. To exclude these parameters from the file, you can use the IGNORE IN parameter in the IOALDNRS utility, which is described in the *INCONTROL for z/OS Utilities Guide*.

## Parameter Prompting Facility—Type 2

The Parameter Prompting facility – Type 2 is a manual job scheduling facility that provides automatic prompting for AutoEdit parameter values. On a given day, the user selects the scheduling tables for execution. The user is then automatically prompted for parameter values required for the execution of the jobs scheduled to run on the specified date.

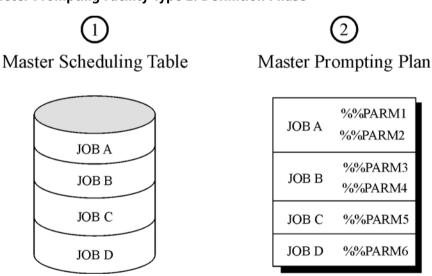
This type of prompting facility is recommended for use in a distributed environment where user departments are responsible for manually scheduling (ordering) their jobs, and specifying required parameters.

A sample application of this type of scheduling facility is the maintenance of confidential salary information in a payroll department. The payroll department usually retains control over its jobs and their input parameters.

The Parameter Prompting facility—Type 2 has three major phases:

#### **Definition Phase**

Figure 361 Parameter Prompting Facility Type 2: Definition Phase



#### 1. Scheduling Table

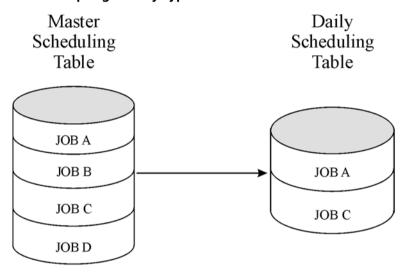
First, a scheduling table is defined using the CONTROL-M Online facility. The scheduling table contains job scheduling information for all of a department's jobs. Any number or type of jobs with any valid date scheduling criteria can be designated. The table is placed in a Master Scheduling Tables library.

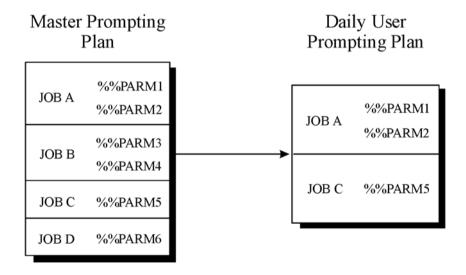
#### 2. Master Prompting Plan

Next, using the first option of the Parameter Prompting facility – Type 2, a Master Prompting Plan (MPP) is defined containing all AutoEdit variables used by all the jobs in the scheduling table. Any default values can be assigned. Value validity checks can also be defined. The MPP is placed in the Master Prompting Plan library.

#### **FETCH Phase**

Figure 362 Parameter Prompting Facility Type 2: Fetch Phase





The second option of the Parameter Prompting facility – Type 2, FETCH A PLAN, allows the user to select a plan for execution by CONTROL-M on a specific day.

When a FETCH option is executed for a specific PLANID, or all PLANIDs with a specific suffix, a daily scheduling table is automatically created. The Daily Scheduling table, a subset of the Master Scheduling table, is placed in the Daily Scheduling Tables library. The Daily Scheduling table contains the job scheduling definition of all of the jobs in the Master Scheduling table scheduled to run on the specified date, based on each job's scheduling criteria.

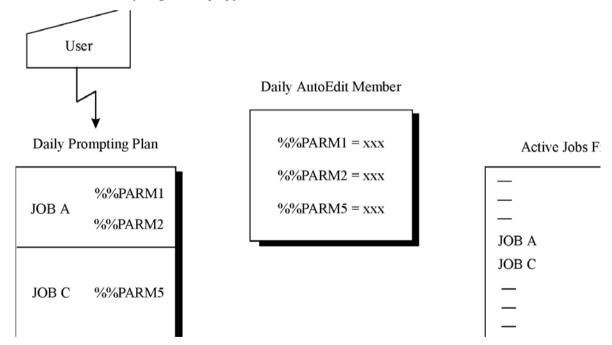
The FETCH also creates a User Prompting Plan (UPP), a subset of the Master Prompting Plan, which is placed in the Daily Prompting Plan library. The UPP contains only parameters that are required by the jobs scheduled to run on the specified date.

A Daily JCL library is also created containing JCL for today's jobs.

#### **EXEC Phase**

The third option of this facility, EXEC A PLAN, permits the user to update or accept the default values of all the parameters appearing in the daily UPP. A daily AutoEdit parameters member, which is accessed at the time of job submission, is automatically created and placed in the Daily AutoEdit Parameter library.

Figure 363 Parameter Prompting Facility Type 2: EXEC Phase



Once values have been assigned to all the parameters, the prerequisite condition RUN-%%PLANID is added. %%PLANID is resolved to the PLANID, and suffix if supplied, designated in the FETCH phase.

The Daily Scheduling table is then ordered by CONTROL-M. The jobs are placed on the Active Jobs file and run based on their scheduling parameters; that is, a job is submitted only when all scheduling criteria, such as other prerequisite conditions, have been met.

## **Tailoring to User Needs**

The Parameter Prompting facility – Type 2 is usually activated from the CONTROL-M ISPF Utilities screen. However, it is possible to activate the FETCH and EXEC phases using the following CLISTS:

**Table 250 Parameter Prompting Facility Type 2: Use of CLISTS** 

CLISTS	Purpose
CTMFETCH	For fetching a plan (FETCH phase)
CTMEXEC	For executing a plan (EXEC phase)

#### CTMFETCH CLIST

When CLIST CTMFETCH is activated, the FETCH A PLAN screen is displayed:

Figure 364 The FETCH A PLAN Screen

```
CONTROL-M P.P.F. FETCH A PLAN (P.2)
COMMAND ===>
  PLAN NAME
                      ===>
  PLAN NAME SUFFIX ===>
                               (For multiple plans in the same day)
  OVERRIDE DAILY PLAN ===> NO
                                   (YES / NO)
  ODATE
                  ===> 060601
 Please fill in the Plan Name and press ENTER
  MASTER SCHEDULING LIB ===> CTMP.PROD.SCHEDULE
  DAILY SCHEDULING LIB ===> CTMP.PROD.SCHD
  MASTER PLANS LIB ===> CTMP.PROD.PLANMSTR
  DAILY PROMPT PLANS LIB ===> CTMP.PROD.PLAN
  MASTER JCL LIB ===> CTMP.PROD.JCLPROMP
  DAILY JCL LIB
                      ===> CTMP.PROD.JCLP
ENTER END COMMAND OR PF3 TO TERMINATE
```

Table 251 The FETCH A PLAN Screen: Parameters

Parameter	Description	
PLAN NAME	Plan name (1 through 6 characters). Mandatory.	
PLAN NAME SUFFIX	Two character suffix to be added to the PLAN NAME in daily libraries. Changing the suffix permits multiple daily plans.	
OVERRIDE DAILY PLAN	Whether to replace an existing plan.  Valid values are:  ■ YES – a duplicate fetch of a plan (with suffix, if one has been designated) replaces an existing copy of a plan with the same PLAN NAME (and same suffix) for that day  ■ NO – multiple fetches of a plan are not permitted on the same day (default)	
ODATE	Specific date for which the plan is to be fetched. Default is the current working date. Another date can be specified, in <i>mmddyy</i> , <i>ddmmyy</i> or <i>yymmdd</i> format, depending on the site standard.	
MASTER SCHEDULING LIB	Name of the Master Scheduling Tables library.	
DAILY SCHEDULING LIB	Name of the Daily Scheduling Tables library. The last qualifier of the library name cannot exceed 4 characters. The CLIST concatenates the date to this daily library name.	
MASTER PLANS LIB	Name of the Master Prompting Plans library.	
DAILY PROMPT PLANS LIB	Name of the User Daily Prompting Plans library. The last qualifier of the library name cannot exceed 4 characters. The CLIST concatenates the date to this daily library name.	
MASTER JCL LIB	Name of the Master JCL library.	
DAILY JCL LIB	Name of the Daily JCL library. The last qualifier of the library name cannot exceed 4 characters. The ** symbol concatenates the date to this daily library name.	

#### **CTMEXEC CLIST**

When CLIST CTMEXEC is activated, the EXEC / ORDER A PLAN screen is displayed:

Figure 365 The EXEC / ORDER A PLAN Screen

```
---- CONTROL-M - P.P.F. ---- EXEC / ORDER A PLAN ------(P.3)
COMMAND ===>
   PLAN NAME
                      ===> REPTS (Blank for plan selection list)
   PLAN NAME SUFFIX ===>
                                     (For multiple plans in the same day)
   REMAINING PARAMETERS ===> NO
                                    (YES / NO)
                       ===> 080800
   ODATE
   FORCED FROM TIME
                      ===>
 Please fill in the Plan Name (or blanks) and press ENTER
   DAILY SCHEDULING LIB ===> CTM.PROD.SCHD
   USER PROMPT PLANS LIB ===> CTM.PROD.PLAN
   DAILY PARAMETERS LIB ===> CTM.PROD.AEDI
ENTER END COMMAND OR PF3 TO TERMINATE
```

Table 252 The EXEC / ORDER A PLAN Screen: Parameters

Parameter	Description
PLAN NAME	Plan name (1 to 6 characters). Mandatory.
PLAN NAME SUFFIX	2-character suffix used to specify a specific plan.
REMAINING PARAMETERS	<ul> <li>Continuation instructions. Valid values are:</li> <li>Y - The user is automatically presented with remaining (non-updated) parameters from all active plans</li> <li>N - After updating the current plan, the user is given options for choosing another plan (default)</li> </ul>
ODATE	Specific date on which the plan is ordered. Default is the current working date. Another date (in mmddyy, ddmmyy, or yymmdd format depending on the site standard) can be specified.
FORCED FROM TIME	Specific time in format hhmm, before which the jobs cannot run.
DAILY SCHEDULING LIB	Name of the Daily Scheduling Tables library.
USER PROMPT PLANS LIB	Name of the User Prompting Plans library.
DAILY PARAMETERS LIB	Name of the Daily Parameters library.

## **Usage Notes**

- The PLAN NAME can be up to six characters in length. Use of the SUFFIX parameter in the FETCH phase permits creation of multiple Daily User Prompting Plans based on one Master Prompting Plan. This also makes it possible to duplicate (by overriding) a fetch of the same plan by setting the OVERRIDE DAILY PLAN parameter to YES on the FETCH A PLAN screen.
- Each job defined in the Master Scheduling table must contain an IN prerequisite condition in the format:

```
RUN-%%PLANID
```

This prerequisite condition is added during the EXEC phase only after all parameters for a specific plan are assigned in the EXEC phase. %%PLANID resolves to the PLAN NAME (and SUFFIX) designated in the FETCH phase.

Since each plan can be ordered multiple times for the same scheduling date, it is highly recommended to distinguish between the dependencies of the jobs in the plan based on PLAN NAME. Every prerequisite condition used for inter-plan dependency must contain the string %%PLANID. It is automatically replaced by the PLANID during the FETCH phase.

■ The JCL of each job must be modified as follows:

The first AutoEdit control statement must point to an AutoEdit Parameters library and the PLANID member. The PLANID member contains the unique PLANID of the job (automatically handled by CONTROL-M).

#### **Example**

```
%%LIBSYM CTM.PROD.SYMBOLS %%MEMSYM PLANID
```

The second AutoEdit control statement must point to the Daily AutoEdit Parameters library and the member %%PLANID. CONTROL-M automatically resolves %%PLANID to the PLAN NAME designated in the FETCH phase.

The Daily AutoEdit Parameter library must be suffixed by a date parameter that is resolved by CONTROL-M. Example:

```
%%LIBSYM CTM.PROD.AEDI%%OMONTH.%%ODAY %%MEMSYM %%PLANID
```

■ The %%PLANID member of each plan (in the Daily AutoEdit Parameter library) contains up to four configuration tables identified by AutoEdit variables.

Parameters and data that are repeatedly used in a data processing environment can be defined in such a configuration table.

A configuration table is a member of the GLOBAL library, which is referenced by the DAGLOBAL DD statement. Such a member contains a set of user-defined local variables and their assigned values that can be referenced in the JCL of individual jobs.

An AutoEdit variable identifies such a configuration table by a statement in the form %%CONF*n*=*config_tablename* in the %%PLANID member.

#### **Example**

```
%%CONF1=INDICES
%%CONF2=WEEKCHAR
%%CONF3=
%%CONF4=
```

To use the configuration tables parameters procedure, insert the following AutoEdit statement in the JCL of each job:

```
//* %%INCLIB CTM.PROD.PARM %%INCMEM PPF2CONF
```

The PPF2CONF member uses the configuration table values specified in the %%PLANID member to select the GLOBAL autoedit members to be included in the JCL of the job. It contains the following AutoEdit code:

```
%%IF *%%CONF1 NE *
%%GLOBAL %%CONF1
%%ENDIF
%%IF *%%CONF2 NE *
%%GLOBAL %%CONF2
%%ENDIF
%%IF *%%CONF3 NE *
%%GLOBAL %%CONF3
%%ENDIF
%%IF *%%CONF4 NE *
%%GLOBAL %%CONF4
%%ENDIF
```

■ The MEMLIB parameter of each job scheduling definition in the scheduling table must point to the Daily JCL library. The library name is suffixed by the AutoEdit date variables.

#### **Example**

```
CTM.PROD.JCLP%%OMONTH.%%ODAY
```

#### ■ Occurrence numbering:

It is recommended that all AutoEdit parameter names of jobs in the same plan be unique. In some plans, duplicate names may be unavoidable, and more than one job may share the same AutoEdit parameter name. If the parameters are to be assigned different values, that is, used for different purposes, each parameter must be assigned a different OCCUR NO during definition of the Master Prompting Plan.

A %%SET statement specifying the OCCUR NO. must be included in the JCL of the associated jobs as follows:

```
\%\%SET \%\%0C\# = 01
```

When the AutoEdit parameter member is created, each AutoEdit parameter includes the OCCUR NO. as a suffix.

- Using the Parameter Prompting facility Type 2 requires customizing the CONTROL-M submit exit (CTMX002). This exit does the following:
  - It ensures that at the time when the job is submitted, the AutoEdit Parameters library contains the PLANID member.
  - It places in the PLANID member an AutoEdit variable definition in the form

```
%%PLANID=plan id
```

in order to provide the job with a unique identity (*plan_id*). This is done using an OUT condition in the job scheduling definition, as described in the source code of the exit.

For information about modifying the exit, see the CTMX2PPF member in the IOA SAMPEXIT library.

## **Maintenance Utilities**

The following jobs are located in the CONTROL-M JCL library.

#### **PPF2DEL**

This job can be run to delete sets of daily libraries according to specified date criteria.

#### Figure 366 PPF2DEL Utility Screen

```
//PPF2DEL JOB ACCNT,CTM,CLASS=A
//*
     THIS JOB DELETES SETS OF DAILY LIBRARIES CREATED 3.4. and 5
    DAYS PRIOR TO THE CURRENT DATE (%%ODATE).
//***********************************
  %%SET %%DT3 = %%CALCDATE %%ODATE - 3
  %%SET %%DELDATE3 = %%SUBSTR %%DT3 3 4
  %%SET %%DT4 = %%CALCDATE %%ODATE - 4
  %%SET %%DELDATE4 = %%SUBSTR %%DT4 3 4
  %%SET %%DT5 = %%CALCDATE %%0DATE - 5
  %%SET %%DELDATE5 = %%SUBSTR %%DT5 3 4
//DELLIB EXEC PGM=IDCAMS
//SYSPRINT DD
               SYSOUT=*
//SYSIN
         DD
  DELETE CTM.PROD.PLAN%%DELDATE3
  DELETE CTM.PROD.SCHD%%DELDATE3
  DELETE CTM.PROD.JCLP%%DELDATE3
  DELETE CTM.PROD.AEDI%%DELDATE3
  DELETE CTM.PROD.PLAN%%DELDATE4
  DELETE CTM.PROD.SCHD%%DELDATE4
  DELETE CTM.PROD.JCLP%%DELDATE4
  DELETE CTM.PROD.AEDI%%DELDATE4
  DELETE CTM.PROD.PLAN%%DELDATE5
  DELETE CTM.PROD.SCHD%%DELDATE5
  DELETE CTM.PROD.JCLP%%DELDATE5
  DELETE CTM.PROD.AEDI%%DELDATE5
```

#### **PPF2DAY**

This job allocates the daily libraries that are to be used by the Parameter Prompting facility – Type 2. It also copies the required jobs from the Master JCL library to the Daily JCL library.

These are time consuming tasks normally performed as part of the FETCH and EXEC phases of the Online facility. By scheduling this job daily under CONTROL-M, the time required to execute the FETCH and EXEC phases is reduced.

Chapter





# **Simulation and Forecasting Facility**

This chapter includes the following topics:	
Overview	340
Simulation Procedure CTMSIM	340
Activating the Procedure	341
Preparatory Steps 8	342
Parameters 8	
Input Files 8	345
Output Files 8	
CONTROL-M Exits and Simulation Processing	347
Analyzing the Simulation Run 8	
Handling Manual Conditions 8	350
The CTMTAPUL Tape Pull List Procedure	351
Activating the Procedure 8	
Parameters	
DD Statements	355
JOB/SCAN, PRO/JCL, DOCU/TEXT Interface	355
Problem Determination for Tape Pull List Reports 8	356
Sample Tape Pull List Reports	

## **Overview**

The Simulation and Forecasting facility consists of two components.

- Simulation procedure CTMSIM
- Tape Pull List procedure CTMTAPUL

Simulation procedure CTMSIM tests the potential impact of proposed changes to the configuration or production environment. It answers "What if?" questions, such as:

"What if we ...

- add or remove four tape drives from the system?"
- increase the CPU power by 30%?"
- run a particular set of applications daily instead of weekly?"

The Simulation procedure can also be used to forecast production runs such as the next 24-hour run, end-of-month run, and so on. In this way, possible irregularities in the schedule can be foreseen.

The CTMTAPUL Tape Pull List procedure creates a report of all tapes to be mounted for a specific period. By running the procedure on a daily basis, all tapes required for the daily production run can be prepared in advance for the operator or the robotic tape library.

The Tape Pull List procedure uses the output of the Simulation procedure as input. Therefore, the Simulation procedure must be executed before the Tape Pull List procedure can be executed. However, the Simulation procedure can be executed without executing the Tape Pull List procedure.

# **Simulation Procedure CTMSIM**

The Simulation procedure mirrors the CONTROL-M monitor flow for a specified period without actual job submission and without output processing. It takes into consideration all scheduling criteria including prerequisite conditions, time limits, quantitative, and Control resources.

CONTROL-M and IOA files used as input to the simulation process are not updated as a result of this process. The statuses of the files after simulation are written to special simulation output files. The simulation assumes that each job ended execution with a condition code of 0.

The following CONTROL-M and IOA files are used as input in the Simulation procedure. These files may either be actual production files or special files created specifically for forecasting purposes.

Table 253 Files Used as Input during Simulation

File	Description
AJF	Active Jobs file
RES	CONTROL-M Resources file
CND	IOA Conditions file
STAT	CONTROL-M Statistics file

The simulation produces the following output files:

**Table 254** Files Produced as Output of Simulation

File	Description
SIMOAJF	Simulation Output Active Jobs file
SIMORES	Simulation Output Resources file
SIMOCND	Simulation Output Conditions file
SIMLOG	Simulation Output Log file

The Simulation procedure may contain several steps prior to the actual simulation processing step, depending on the environment to be used as input to the simulation run. For example

- It may be desirable to use the production Active Jobs file as input.
- It may be necessary to simulate the run of a specific day without relating to the current production jobs.
- It may be necessary to resolve manual conditions prior to the simulation run so that all jobs are submitted.

## **Activating the Procedure**

The procedure can be invoked directly, that is, through // EXEC CTMSIM, or through a job generated by option M3 of the Online Utilities menu.

Option M3 is the preferred method of generating the job stream for the procedure because the Online utility can automatically perform certain necessary preparatory steps (mentioned above). If the Online utility is not used, the user must add these steps, as necessary, to the JCL.

The M3 Online utility is discussed in "M3: Prepare Simulation/Tape Pull List Job" on page 332.

## **Preparatory Steps**

The following preparatory steps must be added, as needed, to the simulation job.

#### **Allocate IOA Conditions File**

The DACNDF DD statement references the IOA Conditions file. It may refer to either of the following files:

- the production IOA Conditions file If this file is used, it only needs allocation.
- a test file used to simulate jobs planned for a future working day If a test file is used, an initial job step must allocate and format the simulation Active Jobs file. A date record is then generated for the date of the simulation. A User Daily job step then loads the required job orders into the simulation Active Jobs file.

This preparatory step can be automatically generated through the M3 online utility.

All jobs in the Active Jobs file participate in the simulation. For jobs that are currently executing at the time the simulation is running, it is assumed that they have already executed half of their average elapsed time.

#### Allocate CONTROL-M Resources File

The DACNDF DD statement references the CONTROL-M Resources file. It may refer to either of the following files:

- the production CONTROL-M Resources file If this file is used, it only needs allocation.
- a test file, which can be used to define Quantitative resources under simulation, using the IOACND utility

## **Parameters**

There are two ways of setting parameters for the utility.

Parameters can be passed to the utility using the SIMPARM member in the CONTROL-M PARM library. This member is referenced by the DASIMPRM DD statement in the Simulation procedure, if it is specified in option M3 (Prepare Simulation/Tape Pull List Job) of the Online Utilities.

Alternatively, parameters may be passed to the utility in-line, using the DASIMPRM (or SYSIN) DD statement.

Table 255 Parameters Passed to the Utility by DASIMPRM (part 1 of 3)

Parameter	Description	
SIMSTART	Date and time at which the simulation must start, in yymmddhhmm format. Mandatory.	
SIMEND	Date and time at which the simulation must end, in yymmddhhmm format. Mandatory.	
	Note: Ordinarily the interval between specified SIMSTART and SIMEND values should not exceed a 24 hour period because there is no mechanism to simulate New Day processing for the next day. However, it is possible to specify a larger interval if one is required to enable existing jobs to complete.	
INTERVAL mm	Simulation interval, in minutes. The simulation "clock" advances by the interval specified. The shorter the interval, the more accurate the simulation, but the longer the simulation takes to run. The specified interval must not exceed one working day. Mandatory.	

Table 255 Parameters Passed to the Utility by DASIMPRM (part 2 of 3)

Parameter	Description
NEWJOB	For a job that has no execution statistics, this statement is used to indicate the expected execution time of the job. Optional. If the simulation encounters a job without statistics and this statement is not supplied, a default execution time of three minutes is used.  The format of the NEWJOB parameter is as follows:  NEWJOB memname EXECTIME mmmm.xx [GROUP groupname][CPUID i]  The following subparameters can be specified:  memname – name of the member containing the JCL of the job. This value helps identify the job order in the Active Jobs file. Mandatory.  mmmm.xx – expected execution time, where mmmm is the number of minutes and xx is hundredths of minutes. This is the same format used in the sysout Log message of the job. Mandatory.  groupname – name of the group to which the job belongs. This value helps identify the job order in the Active Jobs file. Optional.  i – CPU ID. In a multiple CPU environment, the job can have different execution times on different CPUs. Therefore, it is useful to specify the expected elapsed time for each CPU on which the job may run. i must have the same value as \$SIGN, which is described in "%%\$SIGN" on page 750. Optional.
OLDJOB	For a job with execution statistics, this statement can be used to override the statistically estimated execution time. For example, a longer execution time can be specified to test the effect of adding more input data to the job. Optional.  Format of the OLDJOB parameter is as follows:  OLDJOB memname EXECTIME mmmm.xx [GROUP groupname][CPUID i]  The same subparameters can be specified for the OLDJOB parameter as those for the NEWJOB parameter (in this table).
ADD	Add or delete a prerequisite condition of a specific ODATE (original scheduling date) at a specified simulation date and time.
or	The format of the ADD or DELETE parameter is as follows:  {ADD DELETE} COND condname odate ONDAYTIME yymmddhhmm
DELETE	<ul> <li>The following subparameters can be specified:</li> <li>ADD   DELETE – action to be performed. Mandatory.</li> <li>condname – name of the condition to be added or deleted.         Mandatory.</li> <li>odate – original scheduling date associated with the condition.         Mandatory.</li> <li>yymmddhhmm – simulation date and time at which the condition must be added or deleted. Mandatory.</li> </ul>

Table 255 Parameters Passed to the Utility by DASIMPRM (part 3 of 3)

Parameter	Description
CHANGE RESOURCE	Change the quantity of a given resource at a specified simulation date and time.  Format of the CHANGE RESOURCE parameter is as follows:  CHANGE RESOURCE resname quantity ONDAYTIME yymmddhhmm  The following subparameters can be specified:  resname – name of the resource whose quantity is to be changed.  Mandatory.  quantity – change in quantity for the resource. The quantity change can be specified in any of the following formats:  nnnn – set the quantity to the specified value  +nnnn – add the specified quantity to the current quantity  -nnnn – subtract the specified quantity from the current quantity  yymmddhhmm – simulation date and time at which the resource quantity must be changed. Optional.

## **Input Files**

The simulation procedure accepts the following input files:

- Active Jobs file—created in the course of the preparatory steps described in "Preparatory Steps" on page 842
- Job Execution Statistics file—the DASTAT DD statement references the CONTROL-M Job Execution Statistics file; this file contains job execution statistics, including the execution elapsed time
- IOA Conditions file—created in the course of the preparatory steps described in "Preparatory Steps" on page 842
- CONTROL-M Resources file—created in the course of the preparatory steps described in "Preparatory Steps" on page 842

## **Output Files**

The simulation run produces the following output files:

■ Simulation Messages file

The DASIMOUT DD statement references a sequential file containing a list of the simulation parameters used, and special messages for error codes, warning situations, and so on. In addition, it contains all the SHOUT messages to TSO/ROSCOE users and to the computer operator.

■ SIMLOG—Simulation Log file

The DALOGOUT DD statement references a sequential output file in a format similar to that of the IOA Log file.

At the end of the simulation process, this file contains all the log messages describing events that occurred during the simulation run, for example, JOB SUBMITTED, JOB ENDED OK, or COND xxxx ADDED.

This file can be used as input to all CONTROL-M reports that are normally produced from the IOA Log file. The standard set of reports produced from the IOA Log can be used to analyze the simulation run. Therefore, it is not necessary to write special simulation reports.

■ SIMOAJF—Active Jobs file

The DACKPTOU DD statement references a file in the format of the Active Jobs file. All jobs that were present in the input Active Jobs file are written to this file with the status assigned to them during the simulation run, for example, ENDED OK, EXECUTING, or WAIT SCHEDULE.

The file can be scanned online or in batch mode using standard CONTROL-M facilities.

■ SIMOCND—Conditions file

The SIMOCND DD statement references a file in the format of the IOA Conditions file. All conditions that were present at the end of the simulation run are written to this file.

The file can be scanned online or in batch mode using standard CONTROL-M facilities.

#### ■ SIMORES—Resources file

The SIMORES DD statement references a file in the format of the CONTROL-M Resources file. All Quantitative resources and Control resources that were present at the end of the simulation run are written to this file.

The file can be scanned online or in batch mode using standard CONTROL-M facilities.

## **CONTROL-M Exits and Simulation Processing**

The CONTROL-M Simulation and Forecasting facility functions in much the same way as the CONTROL-M monitor, but is activated without performing "real I/O." Therefore, some of the exits activated under the CONTROL-M monitor are also activated during simulation.

Exit CTMX003 (output scanning) is invoked once for each job. The exit does not receive any sysout. Since the simulation assumes that each job ended execution with a condition code of 0, this exit can also be used to add events for certain jobs that end with a nonzero condition code that influences the job flow.

Exit CTMX002 is not activated in simulation mode.

If the same exit is to be used in both the production and simulation environments, it may be necessary to determine which environment is currently active. The MCTSMIND field in the MCT can be checked as follows to determine whether the exit is running under simulation:

TM	MCTSMIND,MCTSIMYS	ARE WE UNDER SIMULATION?	
В0	SKIPPROD	YES, SKIP PRODUCTION LOGIC	

## **Sample Input**

#### Figure 367 CONTROL-M Simulation Exit Screen

```
EXEC CTMSIM
//SIM.DACKPTIN DD
                         DSN=XXX.CTM.PROD.TESTAJF,DISP=SHR
//SIM.SYSIN DD
SIMSTART 0106060900
SIMEND 0106062200
INTERVAL 15
NEWJOB D4TRY1 EXECTIME 2.30
NEWJOB D4TRY2 EXECTIME 100.00 CPUID 1
NEWJOB D4TRY2 EXECTIME 120.00 CPUID 2
CHANGE RESOURCE TAPE -3 ONDAYTIME 0106061600
ADD COND IR-TAPE-ARRIVED 0606 ONDAYTIME 0106061800
ADD COND END-CICS 0606 ONDAYTIME 0106062100
    . SIMO76I SIMULATION STARTED
SIMSTART 0112122100
SIMEND 0112122130
INTERVAL 55
ADD COND TAP-TEST-OK 0606 ONDAYTIME 0106062102
ADD COND TAP-TEST2-OK 0606 ONDAYTIME 0106062102
ADD COND PUL2-OK 0606 ONDAYTIME 0106062102
ADD COND PUL1-OK 0606 ONDAYTIME 0106062102
ADD COND PUL2-OK 0606 ONDAYTIME 0106062102
ADD COND PUL2-OK 0606 ONDAYTIME 0106062102
NEWJOB ASMMCTD EXECTIME 0001.00
NEWJOB ASMMCTM EXECTIME 0001.00
21.00.00 RUN100I CONTROL-M MONITOR STARTED
21.00.00 SIMO87W MEMBER PRDJBREG LIBRARY PROD.DAILY.JOBS
DEFAULT ELAPSED TIME USED
21.00.00 SIMO87W MEMBER PRDJBDAY LIBRARY PROD.DAILY.JOBS
DEFAULT ELAPSED TIME USED
                                                ODATE 0606 ADDED
ODATE 0606 ADDED
21.02.45 CTM567I COND PUL2-OK
21.02.45 CTM567I COND PUL1-0K
                                                     ODATE 0606 ALREADY EXISTS
21.02.45 CTM587I COND PUL2-OK
                                              ODATE 0606 ADDED
21.02.45 CTM567I COND TAP-TEST2-OK
21.02.45 CTM567I COND TAP-TEST-OK
                                                       ODATE 0606 ADDED
21.31.10 SIMO98I TASK EXECDAY DID NOT FINISH EXECUTING 21.31.10 SIMO98I TASK PRDTEST DID NOT FINISH EXECUTING
21.31.10 SIM099I TASK MPMXXX STILL WAITS SCHEDULE 21.31.10 SIM099I TASK MPMTST STILL WAITS SCHEDULE
21.31.10 RUN120I CONTROL-M MONITOR SHUTTING DOWN
21.31.10 SIMO78I SIMULATION ENDED
```

## **Analyzing the Simulation Run**

The following tools can be used to analyze the simulation run and to diagnose problems that may have occurred during simulation processing.

- output of the Simulation Run
- output of the KSL Step
- Night Schedule Report
- Online Simulation Environment
- the CTMRAFL utility

These tools are described below.

## **Output of the Simulation Run**

The DASIMOUT DD statement references summary information about the simulation run. It specifies which jobs ran, which are still in WAIT SCHEDULE status, and which are still executing when the simulation terminates. The following tools can be used to ascertain why certain jobs remain in WAIT SCHEDULE status when the simulation run is terminated.

## **Output of the KSL Step**

The REP3LEFT KSL script can be executed after the simulation step. REP3LEFT generates a report that shows the reasons why certain jobs are still in WAIT SCHEDULE status at the end of the simulation run. This report can be requested as an option through the M3 Online utility.

## **Night Schedule Report**

This report provides a summary of each job that fell within the time interval of the simulation run. This report can be requested as an option through the M3 Online utility.

## **Online Simulation Environment**

A special online environment can be created for the allocation of the files written by the simulation run. The online environment must include the following allocations:

**Table 256 Online Simulation Environment File Allocations** 

Allocation	Description
DACKPT DD statement	Allocated to file SIMOAJF
DACNDF DD statement	Allocated to file SIMORES
DACNDF DD statement	Allocated to file SIMOCND
DALOG DD statement	Allocated to file SIMLOG

The online environment can be used to determine not only which jobs were submitted, but which jobs are waiting to be scheduled and why they remained in WAIT SCHEDULE status at the termination of the simulation run.

## The CTMRAFL Utility

The CTMRAFL utility, which is described in the *INCONTROL* for z/OS Utilities Guide, can be run on the simulation input Active Jobs file to obtain information on job dependencies and manual conditions. The CTMRAFL utility does not check the IOA Conditions file. Therefore, conditions listed as "manual conditions" may actually exist in the IOA Conditions file.

## **Handling Manual Conditions**

Perform the following steps to incorporate manual conditions into the IOA Conditions file that is used in the simulation procedure.

- Create a Conditions file and a Manual Conditions file to be used for simulation only. You can use the FORMCND and FORMNRS members in the IOA INSTALL library to do this. The file name CND can be changed to SIMCND. The file names NRS and NSN can be changed to SIMNRS and SIMNSN respectively.
- 2. Using the CTMCOPRS utility, copy the contents of the production IOA Conditions file into the simulation Conditions file created in Step 1.
- 3. Integrate the IOALDNRS utility into the standard CTMSIM procedure so that it runs against the simulation Active Jobs File, which has been loaded with simulation jobs, to load the manual conditions into the simulation NRS file SIMNRS.

Specify the following overrides when using the IOALDNRS procedure:

Table 257 Overrides To Be Specified on IOALDNRS

DDname	DSname Suffix	Override Suffix
DANRES	NRS	SIMNRS
DANSINC	NSN	SIMNSN
DACNDF	CND	SIMCND

4. Integrate the ADDCMND job, which is in the CONTROL-M JCL library, into the standard CTMSIM procedure to add the required manual Maybe conditions to the simulation Conditions file.

Specify the following overrides when running ADDMNCND:

#### Table 258 Overrides To Be Specified on ADDMNCND

DDname	DSname Suffix	Override Suffix
DANRES	NRS	SIMNRS
DANSINC	NSN	SIMNSN

5. Specify the following override for the simulation step:

**Table 259 Override To Be Specified for Simulation Step** 

DDname	DSname Suffix	Override Suffix
DACNDF	CND	SIMCND

# The CTMTAPUL Tape Pull List Procedure

The Tape Pull List procedure creates a list of all tapes to be mounted in a specified period. The list can be sorted and edited in various ways, such as

- all tapes to be mounted, sorted by the expected mount time
- all tapes to be mounted, sorted by volume serial number

#### NOTE



It is highly recommended that the simulation be run from the current time, that is, not from a time in the future. Otherwise, the Tape Pull list results may be inaccurate because new tape files may be cataloged in the time remaining before the start of the simulation run.

The procedure takes into account the expected order of job execution and the order of creation of tape data sets.

The procedure also does the following:

- checks the syntax of all AutoEdit statements in all jobs that are planned for the given period
- checks the JCL syntax
- produces a list of data sets that are still missing for the execution.

These are usually input data sets due to arrive, but they may be JCL execution errors

#### – NOTE



For the Tape Pull List procedure to be executed properly, the internal reader (INTRDR) must have authority to submit jobs.

The Tape Pull List procedure uses files from the Simulation procedure as input. In preparation for the Tape Pull List procedure, run the Simulation procedure using the production CONTROL-M Active Jobs file, CONTROL-M Resources file, and IOA Conditions file.

The Tape Pull List procedure works as follows:

- The procedure looks for "SUBMISSION" messages in the simulation output LOG file.
  - For each submission message, it looks for the appropriate job having a WAIT SCHEDULE status on the simulation input Active Jobs file.
  - If a job is found, the Tape Pull List procedure actually submits the job with the TYPRUN=SCAN parameter specification, reads the SYSOUT of the job, retrieves the data set information, and produces the required reports.
- The procedure recognizes tape data sets by either the appropriate unit specification in the JCL, such as UNIT=TAPE, or by retrieving information from the system catalog. It is therefore not necessary to have all tape data sets cataloged in the MVS catalog.

#### NOTE -



A highlighted warning message appears on the operator console while the jobs are being submitted. During this stage, the operator console may display several messages regarding the job submission. The highlighted message disappears at the end of this stage.

The procedure can detect that a certain tape is used by more than one job, and which job creates it, as illustrated in the following example.

Job A creates a new generation of a tape data set:

```
//OUTUPD DD DSN=PFX.DATA(+1),DISP=(,CATLG),...
```

Job B, a successor of Job A, accesses the same data set:

```
//INUPD DD DSN=PFX.DATA(0),DISP=SHR
```

The procedure detects that Job A is the creator of the data set used by Job B and reports the same tape volser for both jobs.

■ After the job sysout has been analyzed, the sysout is deleted from spool.

■ The next stage of the procedure produces reports according to the requested parameters.

# **Activating the Procedure**

The Tape Pull List procedure is activated as follows:

```
// EXEC CTMTAPUL
//TAPULIN DD *
parameters
```

#### **Parameters**

There are two ways of setting parameters for the utility.

Parameters can be passed to the Tape Pull List utility using the TAPULPRM member in the CONTROL-M PARM library. This member is referenced by the TAPULIN DD statement in the Tape Pull List procedure, CTMTAPUL, if it is specified in option M3 (Prepare Simulation/Tape Pull List Job) of the Online Utilities.

Alternatively, parameters may be passed to the utility in-line, using the TAPULIN DD statement.

Table 260 CTMTAPUL Subparameters (part 1 of 2)

Subparameter	Description
REPBYVOL	Produce a report sorted by volume serial number (volser). All tapes from the tape library are included.
REPBYTIME	Produce a report sorted by the expected mount time.
REPBYJOB	Produce a report sorted by job name.
REPBYDSN	Produce a report sorted by data set name.

Table 260 CTMTAPUL Subparameters (part 2 of 2)

Subparameter	Description
JCLFILE {YES   NO   ONLY}	Whether a copy of every submitted job is written to the file referenced by the DAJCLOUT DD statement.  Valid values are:  ■ YES – A copy of every submitted job is written to the file referenced by the DAJCLOUT DD statement.  ■ NO – A copy of every submitted job is not written to the file referenced by the DAJCLOUT DD statement. Default.  ■ ONLY – A copy of every submitted job is written to the file referenced by the DAJCLOUT DD statement, but — the procedure does not submit the job — the Tape Pull reports are not created — the procedure is run to create the JCL file only
	Note: When dealing with submitted jobs, the utility, for internal processing purposes, inserts the following accounting code into the jobcard of the job: CTM-FORCE-TPLNM  When operating under JES3, the following statement is also inserted:  //*NET ID=AESUSER
IGNORE DSN dsn	Data set name (or prefix) that must not appear in the report. If the last character of <i>dsn</i> is *, it is treated as a prefix.
IGNORE JOB jobname	Job name (or prefix) that must not appear in the report. If the last character of <i>jobname</i> is *, it is treated as a prefix.

The Tape Pull List job can be prepared using the (ISPF) Simulation panel (Option M3 in the IOA Online Utilities menu). A special section of the panel is designated for Tape Pull List parameters. If you want to run the Tape Pull List procedure

- 1. Type **Y** in the TAPE PULL LIST field.
- 2. Type **Y** in the field next to the desired type of report.

The default control parameters member name appears on the screen. This member contains IGNORE statements (procedure parameters). JOB/SCAN and PRO/JCL parameters are discussed in "JOB/SCAN, PRO/JCL, DOCU/TEXT Interface" on page 855.

After filling in the parameters, enter the edited JCL of a job to run the simulation and the procedure. You can submit it, or save it for future use.

#### **DD Statements**

The following DD statements are used by procedure CTMTAPUL:

Table 261 DD statements used by CTMTAPUL

DD Statement	Description
DALOGIN	Output Log file of the Simulation facility, which is input for the Tape Pull List procedure.
DACKPSOU	Output Active Jobs file of the Simulation facility, which is input for the Tape Pull List procedure
DACKPTIN	Active Jobs file used as input to the simulation, but remains unmodified.
TAPULIN	Procedure parameters.
DATAPNAM	Member containing a list of local names used by the site to describe tape units and cassettes or cartridges.  The TAPNAM member in the CONTROL-M PARM library contains a sample list of local names used by the data center to describe tapes, cassettes, cartridges and DASD units. CONTROL-M recognizes IBM-supplied names such as 3490, which do not need to be specified. This member is referenced by this DATAPNAM DD statement in the Tape Pull List procedure, CTMTAPUL.  The format of each line in the list is:  ■ Columns 1 through 8 − Unit name  ■ Column 9 − One of the following indicators:  — T − Tape  — C − Cassette or Cartridge  — D − DASD
DAREPOUT	Reports output.
TAPULOUT	Messages (of the procedure).
DAJCLOUT	Job stream of the jobs that is submitted during the specified period. For more information, see the following section, "JOB/SCAN, PRO/JCL, DOCU/TEXT Interface."

## JOB/SCAN, PRO/JCL, DOCU/TEXT Interface

When the JCLFILE parameter is specified, every job that is submitted by the procedure for JCL scan is also written to the DAJCLOUT DD statement. At the end of execution of the procedure, this data set (if allocated) contains all jobs that are submitted during the execution period according to the order of submission. This file can be used as input to JOB/SCAN, PRO/JCL, or DOCU/TEXT.

In sites using the JOB/SCAN, PRO/JCL, DOCU/TEXT Interface, the lower portion of the M3 Online utility, which is described in "M3: Prepare Simulation/Tape Pull List Job" on page 332, contains INVOKE JOB/SCAN parameters.

## **Problem Determination for Tape Pull List Reports**

The following conditions must be satisfied before the Tape Pull List procedure can forecast tapes for a specific job:

- The submission message must be present in the simulation output Log file.
- The corresponding job has a WAIT SCHEDULE status in the simulation input Active Jobs file.

Reports are not produced if one or more of the following situations exist:

- None of the "submitted" jobs required tapes.
- Jobs requiring tapes were not "submitted" by the simulation because their submission criteria were not satisfied.
- An invalid Active Jobs file was specified as an input for the Tape Pull List procedure.
- An invalid Log file was specified as an input for the Tape Pull List procedure.
- JCLFILE ONLY was specified as an input parameter for the procedure.
- No reports were requested from the procedure through the input parameters, that is, the default was "no reports".
- The procedure does not recognize tape data sets. For more information, see the description of the Tape Pull list utility in the discussion of CONTROL-M customization in the INCONTROL for z/OS Installation Guide.

# **Sample Tape Pull List Reports**

The following pages show a series of samples of reports produced by CTMTAPUL. In these samples, when the values M-N or M-O appear in the DISP (disposition) column, they signify a disposition of MOD, either as a NEW (M-N) or OLD (M-O) data set.

Figure 368 Sample Tape Pull List Report 1

PRODUCED BY CONTROL-M PROD TAPE PULL LIST BMC SOFTWARE, INC. ====================================			
JOBNAME USER ODATE TIME MEMNAME VOLSER LAB# TYPE DISP DSNAME	PROCNAME	STEPNAME	DDNAME
SMFSAVE M05 060601 09:06 SMFSAVE 110050 0001 T M-0 BKP.MONTH.CONT02(+0)		STEP3	TAPE1
110048 0001 T M-O BKP.MONTH.CONT02(-1) 110049 T		STEP3	TAPE2
110058 0001 T M-O BKP.MONTH.CONTO2(-2) 110059 T		STEP3	TAPE3
PRDINPUT M05 060601 09:02 PRDINPUT 996713 0001 T NEW PRD.TP.FILE1(+1)  S#0001 0001 C NEW PRDW.FILE.GDG.GRPM1(+1)  100047 0001 T M-O BKP.WEEK.CONT01(-2)		ST1 BADSTEP ASTEP1	DD1 BADD1 OUTO
PRDUPDT1 M05 060601 09:09 PRDUPDT1 114003 0001 T OLD PRDW.FILE.GDG.GRP11(+0) 114002 0001 T OLD PRDW.FILE.GDG.GRP11(-1)		RES RES	TAP1GDG0 TAP1GDG2
S#0004 0001 T NEW BKP.MONTH.CONTO1(+1) 114003 0001 T OLD PRDW.FILE.GDG.GRP11(+0) \$#0005 0001 C NEW PRDO.TP.UPDT1		CYCLIC	DACYCT TAPEGDGO CASSFILE
			OUT DACYCT
\$#0007 0001 T NEW BKP.MONTH.CONTO2(+1) PRDRPT2D M05 060601 09:22 PRDRPT2D 114003 0001 T OLD PRDW.FILE.GDG.GRP11(+0) T00001 0001 T NEW PRD.SNG1912.TAPE1(+1)		CYCLIC ST1 ST1	DACYCT TAPEO TAPE1
T00002 0002 T NEW PRD.SNG1912.TAPE1(+2) 994529 0001 T NEW PRDW.FILE.GDG.GRP21(+0)		ST1 ST2	TAPE11 TAPE0
997892 0003 T NEW PRD.SNG1912.TAPE2(+1) S#0008 0001 T NEW &&NEWTEMP 996638 T NEW BKP.MONTH.CONT01(+3)		ST2 ST2 ST2	TAPE2 TAPETMP TAPE802
PRDEXE2E M05 060601 09:25 PRDEXE2E T00002 0002 T		STEP1 STEP1	INTAPE1 INTAPE11
S#0007 0001 T		STEP2 STEP2	TAPE5 TAPE6
110048 0001 T OLD BKP.MONTH.CONT02(-2) 110049 T		STEP2	TAPE7
110058 0001 T OLD BKP.MONTH.CONTO2(-3) 110059 T		STEP2	TAPE8
110056 0001 T OLD BKP.MONTH.CONTO2(-4) 110057 T		STEP2	TAPE9
SORTBY JOBNAME (REPBYJOB)			

# Figure 369 Sample Tape Pull List Report 2

PRODUCED BY CONTROL-M PROD		TAPI	E PUI	L I	IST			
BMC SOFTWARE, INC. DSNAME	JOBNAME	MEMNAME	VOLSER	DISP	TYPE	PROCNAME	STEPNAME	DDNAME
& & N F W T F M P	PRDRPT2	D PRDRPT21	 D S#0008	B NEW	T		ST2	TAPETMP
BKP.MONTH.CONTO1(+1)	PRDUPDT1	PRDUPDT1	S#0004	NEW	Τ	RESTORE	CYCLIC	DACYCT
BKP.MONTH.CONTO1(+3)	PRDRPT2D	PRDRPT2D	996638	NEW	Τ		ST2	TAPE802
BKP.MONTH.CONTO1(+0)	SMFSAVE	SMFSAVE	110050	M - 0	Τ		STEP3	TAPE1
			110051				STEP3	TAPE1
	PRDEXE2E	PRDEXE2E	S#0007	OLD	Τ		STEP2	TAPE5
BKP.MONTH.CONTO2(+1)	PRDRPT2C	PRDRPT2C	S#0007	NEW	Τ	BACKUP	CYCLIC	DACYCT
BKP.MONTH.CONTO2(-1)		SMFSAVE					STEP3	TAPE2
		PRDEXE2E					STEP2	TAPE6
			110051	OLD	Т		STEP2	TAPE6
BKP.MONTH.CONTO2(-2)	SMFSAVE	SMFSAVE					STEP3	TAPE3
			110059				STEP3	TAPE3
	PRDFXF2F	PRDEXE2E					STEP2	TAPE7
			100049				STEP2	TAPE7
BKP.MONTH.CONTO2(-3)	PRDFXF2	E PRDEXE2					STEP2	TAPE8
BK( : 110K(111:00K(102( 0)	TRUENCE	LINDEXEE	110059				STEP2	TAPE8
BKP.MONTH.CONTO2(-4)	PRDFXF2F	PRDEXE2E					STEP2	TAPE9
DKI :1104111:004102( 1)	TRUENCEL	TRUENCEL	110057				STEP2	TAPE9
BKP.WEEK.CONTO1(-2)	PRDINPIIT	PRDINPUT					ASTEP1	OUTO
PRD.SNG1912.TAPE1(+0)		PRDEXE2E					STEP1	INTAPE1
DDD SNG1012 TADE1(±1)	DDDDDT2D	DDDDDT2D	T00001	NEW	т		ST1	TAPE1
PRD SNG1912 TAPE1(+2)	PRDRPT2D	PRDRPT2D	T00001	NEM	T		ST1	TAPE11
PRD.SNG1912.TAPE1(+1) PRD.SNG1912.TAPE1(-1) PRD.SNG1912.TAPE2(+0)	DDDEYE2E	DDDEYE2E	T00002	ULD	T			INTAPE11
DDD SNG1912 TADE2(+0)	DDDEYE2E	DDDEYE2E	007802	OLD	T		STEP2	INTAPE2
PRD.SNG1912.TAPE2(+1)	DDDDDT2D	PRDRPT2D	007802	NEM	T		ST2	TAPE2
DDD TD EILE1/.1)	DDDTMDUT	DDDTNDUT	006712	NIELI	т		ST1	DD1
PRDW.FILE.GDG.GRPM1(+0)	DDDINFUI	DDDINDIIT	540001	ULD	(			BADD2
PRDW.FILE.GDG.GRPM1(+1)	DDDINFUI	DDDINDIIT	S#0001	NEM	C		BADSTEP	
PRDW.FILE.GDG.GRPM8(+1)		PRDINPUT					ASTEP1	TAPEGDG8
PRDW.FILE.GDG.GRP11(+0)		PRDINPUT					ASTEP1	TAPEGDGO
rnuw.iile.dud.dkrii(+U)		PRDUPDT1					RES	TAP1GDG0
		PRDRPT2D						TAPEO
	FRURFIZU	FKUKFIZU	T00002				ST1 ST1	TAPE12
DDDW EILE CDC CDD11(+1)	DDDIMDUT	PRDINPUT						TAPEGDG2
PRDW.FILE.GDG.GRP11(+1)								
PRDW.FILE.GDG.GRP11(-0)		PRDINPUT					ASTEP1	TAPEGDG1
		PRDUPDT1					RES	TAP1GDG1
DDDU 5115 0D0 0DD11( 0)		PRDRPT2D					ST2	TAPEO1
PRDW.FILE.GDG.GRP11(-0) SORTBY DSNAME (REPBYDSN)	PKUUPUT1	PRDUPDT1	114002	ULD	1		RES	TAP1GDG2

## Figure 370 Sample Tape Pull List Report 3

PRODUCED BY (	CONTROL-M	PROD		TAPE	Pl	JLL LIST
BMC SOFTWARE	E, INC.			====		
VOLSER TYPE	JOBNAME	ODATE	TIME	DISP	LAB#	DSNAME
C#0001 C		0.00001		NEU	0001	DDDU TECTELE CDC CDDM1/(1)
						PRDW.TESTFILE.GDG.GRPM1(+1)
- "						PRD.TP.TRANS
						PRDW.TESTFILE.GDG.GRPM8(+1)
						BKP.MONTH.CONTO1(+1)
T00001 T	PRDRPT2D					PRD.SNG1912.TAPE1(+1)
T00002 T						PRD.SNG1912.TAPE1(+2)
						PRDW.TESTFILE.GDG.GRPM8(+0)
100047 T	PRDINPUT					BKP.WEEK.CONTO1(-2)
100048 T	PRDINPUT	060601	22:02	M - 0	0001	BKP.WEEK.CONTO1(-2)
110048 T	SMFSAVE	060601	22:06	M - 0	0001	BKP.MONTH.CONTO2(-1)
	PRDEXE2E	060601	22:25	OLD	0001	BKP.MONTH.CONTO2(-2)
110049 T	SMFSAVE	060601	22:06	M - 0	0001	BKP.MONTH.CONTO2(-1)
	PRDEXE2E	060601	22:25	OLD	0001	BKP.MONTH.CONTO2(-2)
110050 T	SMFSAVE	060601	22:06	M - 0	0001	BKP.MONTH.CONTO2(+0)
	PRDEXE2E	060601	22:25	OLD	0001	BKP.MONTH.CONTO2(-1)
	PRDRUN2F	060601	22:28	M - 0	0001	BKP.MONTH.CONTO2(-2)
110051 T	SMFSAVE	060601	22:06	M - 0	0001	BKP.MONTH.CONTO2(+0)
	PRDFXF2F	060601	22:25	OID	0001	BKP.MONTH.CONTO2(-1)
						BKP.MONTH.CONTO2(-2)
996713 T						1 PRD.TP.FILE1(+1)
						PRD.SNG1912.TAPE2(+1)
						PRD.SNG1912.TAPE2(+0)
	Z Z X Z Z Z	000001		0 20	0000	
SORTBY VOLUME	(REPRY)	/OL)				
JUNION VOLUM	- (NLIDI	V U L /				

# Figure 371 Sample Tape Pull List Report 4

BMC SOFTMARE, INC.  ODATE TIME VOLSER TYPE JOBNAME DISP DSNAME  060601 22:02 996713 T	PRODUCED BY CO		)	TAPE	PULL	LIST		
S#0001 C			JOBNAME					
100047 T	060601 22:02	996713 T	PRDINPUT	NEW	PRD.TP.FI	LE1(+1)		
S#0002 T		S#0001 C		NEW	PRDW.FILE	.GDG.GRF	PM1(+1)	
060601 22:06		100047 T		M - 0	BKP.WEEK.	CONTOl(-	2)	
110051 T		S#0002 T		NEW	PRD.TP.TF	RANS		
100048 T	060601 22:06	110050 T	SMFSAVE	M - O	BKP.MONTH	I.CONTO2	(+0)	
110049 T		110051 T						
110058 T		100048 T		M - O	BKP.MONTH	I.CONTO2	(-1)	
110059 T  060601 22:09 114003 T		110049 T						
060601 22:09				M - O	BKP.MONTH	I.CONTO2	(-2)	
114002 T								
S#0004 T	060601 22:09		PRDUPDT1					
114003 T								
S#0005 C  060601 22:19								
060601 22:19							911(+0)	
113492 T								
S#0007 T	060601 22:19		PRDRPT2C				)RTS	
060601 22:22								
T00001 T	0.00001 00 00	- 11						
T00002 T	060601 22:22		PRDRP12D					
994529 T NEW PRDW.FILE.GDG.GRP21(+0) 997892 T NEW PRD.SNG1912.TAPE2(+1) S#0008 T NEW &&NEWTEMP 996638 T NEW BKP.MONTH.CONTO1(+3)  060601 22:25 T00002 T PRDEXE2E OLD PRD.SNG1912.TAPE1(+0) T00001 T OLD PRD.SNG1912.TAPE1(-1) S#0007 T OLD BKP.MONTH.CONTO2(+0) 110050 T OLD BKP.MONTH.CONTO2(-1) 110048 T OLD BKP.MONTH.CONTO2(-2) 110049 T 110058 T OLD BKP.MONTH.CONTO2(-3) 110059 T								
997892 T							. ,	
S#0008 T								
996638 T NEW BKP.MONTH.CONTO1(+3) 060601 22:25 T00002 T PRDEXE2E OLD PRD.SNG1912.TAPE1(+0) T00001 T OLD PRD.SNG1912.TAPE1(-1) S#0007 T OLD BKP.MONTH.CONTO2(+0) 110050 T OLD BKP.MONTH.CONTO2(-1) 110051 T 110048 T OLD BKP.MONTH.CONTO2(-2) 110049 T 110058 T OLD BKP.MONTH.CONTO2(-3) 110059 T							(+1)	
060601 22:25							(+2)	
T00001 T	060601 22.25		DDDEVE2E					
S#0007 T OLD BKP.MONTH.CONTO2(+0) 110050 T OLD BKP.MONTH.CONTO2(-1) 110051 T 110048 T OLD BKP.MONTH.CONTO2(-2) 110049 T 110058 T OLD BKP.MONTH.CONTO2(-3) 110059 T	000001 22:23		PRUENEZE					
110050 T OLD BKP.MONTH.CONTO2(-1) 110051 T 110048 T OLD BKP.MONTH.CONTO2(-2) 110049 T 110058 T OLD BKP.MONTH.CONTO2(-3) 110059 T								
110051 T 110048 T OLD BKP.MONTH.CONTO2(-2) 110049 T 110058 T OLD BKP.MONTH.CONTO2(-3) 110059 T								
110048 T OLD BKP.MONTH.CONTO2(-2) 110049 T 110058 T OLD BKP.MONTH.CONTO2(-3) 110059 T				OLD	DICI :IIOITII	1.0011102	. 1/	
110049 T 110058 T OLD BKP.MONTH.CONTO2(-3) 110059 T				OLD	BKP.MONTH	L.CONTO2	-2)	
110058 T OLD BKP.MONTH.CONTO2(-3) 110059 T				0.25			. = /	
110059 T				OLD	BKP.MONTH	L.CONTO2	(-3)	
110056 T OLD BKP.MONTH.CONTO2(-4)								
		110056 T		OLD	BKP.MONTH	I.CONTO2	(-4)	
110057 T								
SORTBY TIME (REPBYTIM)	SORTBY TIME	(REPRYTIM)						





# The CONTROL-M Application Program Interface (CTMAPI)

This appendix discusses the CONTROL-M Application Program Interface (CTMAPI), including the following topics:

- "Overview" on page 862
- "Environment and Allocations" on page 863
- "Functions", including
  - "1. Order or Force Existing Jobs" on page 865
  - "2. Create, Order, or Force New Tables" on page 868
  - "3. AJF Actions" on page 870
  - "4. Search" on page 872
  - "5. Global Variables" on page 874
  - "6. Conditions and Resources" on page 876
  - "7. CTMAS Active Job Download Filtering" on page 876
- "Conditional Requests and Selection Criteria" on page 879
- "Return Codes" on page 882
- "Conversational Mode using Program" on page 882
- "Input and Output Registers" on page 883
- "CTMBAPI DSECT" on page 884
- "Status Extension" on page 887
- "Order Extension" on page 892
- "AJF Action Extension" on page 895
- "Global Variable Extension" on page 898
- "Quantitative Resource Extension" on page 899
- "Create and/or Order or Force a Table (BLT)" on page 900
- "Replies" on page 902
- "CTMBAPO" on page 902

# **Overview**

The CONTROL-M Application Program Interface (CTMAPI) is an open interface between the application environment and CONTROL-M. CTMAPI enables your application program to interface with CONTROL-M so that you can access services and extract data from CONTROL-M into your own programs.

CTMAPI is open to all application environments. It can be called from the following programs or environments:

- High Level Language or Assembler programs, running under various environments, such as CICS, IMS, or the like
- a batch job or step
- REXX or CLIST

However, not all functions of the API are applicable to all environments.

The API can call the CTMAPI module and pass it requests through either of the following:

- a function (command line) passed to CTMAPI, as
  - a parameter from within a program
  - a parameter using PARM=variable in a JCL Batch step
  - an explicit command coded in a dedicated sequential file pointed to by the DAAPI special DD statement. Note that only bytes 1 through 72 of the records in DAAPI are processed.

**COBOL** examples:

To perform a job order (with FORCE), in the Working-Storage section code:

```
01 WS-CONTROLM-PARMS.
05 PARM-LENGTH PIC 9(4) COMP VALUE 80.
05 PARM-VALUE PIC X(80) VALUE
'ORDER DDNAME=ddname MEMBER=table-name JOB=job-name FORCE'.
```

In the Procedure division code:

```
CALL 'CTMAPI' USING WS-CONTROLM-PARMS.
IF RETURN-CODE IS GREATER THAN 0 THEN GO TO error-routine.
```

 conversational mode (CTMBAPI mode), using an area mapped by CTMBAPI DSECT. It passes the request from an application program to CTMAPI, and the results are returned to the calling program

These methods, functions and conversational mode, are explained in more detail in this appendix.

# **Environment and Allocations**

CTMAPI is a callable Load module that resides in the IOA LOAD library. It is located below the line (RMODE=24), works in 31 bit addressing mode (AMODE=31), and can be called by programs running in any AMODE.

The following requirements must be satisfied before CTMAPI can be called:

- The calling application must have access to the IOA LOAD library, either using STEPLIB or using Linklist.
- The standard IOA DD statement DAPARM must be allocated to the calling address space before calling CTMAPI, and must correctly point to the IOA PARM and IOA IOAENV libraries. This allocation is essential for the correct loading of CTMPARM, IOAPARM and other required parameter members, and to provide the ability to issue messages, dynamically allocate files, and so on.

In addition to the above allocations, each service requires specific data sets to be allocated for successful execution of the service. For example, to successfully order jobs to CONTROL-M, the Active Jobs file (AJF) must be allocated.

CTMAPI relies on IOA dynamic allocation services to allocate the files appropriate to the function, using an ALC member. This means that your program, REXX or batch requires no knowledge of dynamic allocation.

For more information about IOA dynamic allocation and ALC members, see the *INCONTROL for z/OS Administrator Guide*.

You can tailor CTMAPI to allocate the appropriate files in either of the following ways:

■ Let the function itself dynamically allocate the default data sets based on the site standard naming convention (using the default ALC member). Under each function, you can find which ALC member is used to dynamically allocate the necessary files. If you do not require any unusual allocations, this is the recommended method.

■ If you want to use other allocations, you can prepare your own ALC member and pass it to CTMAPI using the standard DAALOCIN DD statement pointing to your own ALC member. If this method is chosen, it is recommended that you use the default ALC member specified in the function as a basis for your own ALC member.

If the caller is not allocated to DAALOCIN DD at the time CTMAPI is called, it is assumed that the default allocations are to be performed. In this case, CTMAPI will dynamically allocate files using the default ALC member.

If CTMAPI is called under the IOA environment, none of the above is applicable. It is assumed that all the necessary files are already correctly allocated, so no dynamic allocation is performed by CTMAPI.

## **Functions**

CTMAPI supports various types of services, but not all of them are supported under all environments. Some of the functions can be executed using existing IOA or CONTROL-M utilities. For example, CTMJOB can be used to order jobs. Other functions, such as the Status request or the Action request, cannot be processed by means of any existing program or utility.

Future enhancements will be provided first to the API rather than to the appropriate utility. BMC Software therefore recommends that you use CTMAPI for all requests, even functions that are supported using other utilities.

The following CTMAPI functions are available:

- order or force existing jobs into CONTROL-M
   This function can currently also be performed using CTMJOB.
- create and/or order or force a new table into CONTROL-M
   This function can currently also be performed using CTMBLT.

- perform AJF actions equivalent to the following options of the Active Environment screen (Screen 3):
  - Hold
  - —Free
  - Delete
  - Undelete
  - Confirm
  - Rerun
  - Restart
  - React
  - Force OK
  - Kill
- search and query the status and other details of jobs in CONTROL-M
- resolve, set, and checkpoint variables in the IOA Variables Database
- control which jobs in the Active Jobs File are to be excluded from download to Control-M/EM

These functions are described in greater detail in this appendix. Differences in calling the service from different environments are also discussed.

IOAAPI, which is described in the *INCONTROL* for *z/OS* Administrator Guide, can be used to perform the following functions:

- CHECK, ADD, or DELETE conditions
- send e-mail messages
- extract records from the IOA Log file

# 1. Order or Force Existing Jobs

The Order function can be used to order or force an existing scheduling table, or selected jobs from an existing table, to CONTROL-M.

This service can be called from any environment, with few differences between environments. The syntax for this service is as follows:

ORDER  $\{DSN=schedlib \mid [DDNAME \mid DD]=dd\}$   $\{MEMBER \mid MEM=table\}$   $[GROUP=group \mid GRP=group]$  [JOB=jobnm]  $[ODATE \mid DATE=date]$   $[ODATEOPT=VAL \mid RUN]$  [FORCE]  $[INTOGRP=grp_rba$   $[DUP \mid NODUP]]$   $[WTO=N \mid Y]$  [SETVAR=%xxx=yyyy] [SELTAG=tag]... [IGNTAG=tag]... [IF  $if_statement]$ ...

#### – NOTE



In this syntax, NODUP is the default in the expression INTOGRP=*grp_rba* [DUP | NODUP].

Any number of SETVAR statements can be specified in one ORDER statement. The %% string is mandatory. The equal sign between the SETVAR name and the SETVAR value must be contiguous.

For a full description of each parameter, see the description of the CTMJOB utility in the *INCONTROL for z/OS Utilities Guide*. The only change from the utility is the syntax of the Ignore or Select Tags and the SETVAR statements. In CTMJOB, the Ignore and Select Tags are coded separately from the Order statement. Under CTMAPI, they should be coded as part of the Order statement, substituting SELTAG for the keyword SELECT and IGNTAG for the keyword IGNORE. The SETVAR statement can only be specified via CTMAPI. Any job scheduling definition ordered as a result of the Order statement is treated as if it contains the SETVAR statement.

The *selection_criteria* and *if_statement* parts of the command are described under "Conditional Requests and Selection Criteria" on page 879.

Specifying WTO=N causes suppression of writing messages to the console.

# **Order or Force under Batch, REXX or CLIST**

When called from a batch job or step or from a REXX or CLIST, the Order statement is specified as one of the following:

a statement in the format

EXEC CTMAPI PARM='ORDER variable'

■ in a sequential file pointed to by the DAAPI DD statement

In this type of call, the SYSPRINT DD statement must be pre-allocated to the step, and the output of CTMAPI written to this file.

A return code is returned in register R15. For a full list of valid return codes, see "Order or Force Return Codes" on page 867.

### **Order or Force Using Program**

When called from a program, the simplest method of requesting a job order is to pass the Order statement to CTMAPI as a standard parameter. Alternatively, you can use the conversational mode of interface, where the CTMBAPI area is passed as the parameter, and fields in it identify the request. This mode, which is described in "Conversational Mode using Program" on page 882, is most useful when the calling program requires a reply to be returned to it, for example, to keep track of the Order ID of ordered jobs.

The following statement is an example of the standard method of executing CTMAPI in batch job and passing it an Order request:

//DAAPI DD *
ORDER DSN=CTM.PROD.SCHEDULE MEM=DEFSCHD1 JOB=JOBA ODATE=ODATE

#### **Order or Force Allocations**

The default ALC member used by the Order service is ALCMJOBP, which allocates the AJF, calendars, CONTROL-M Statistics file and the UNITDEF member. If you choose to prepare your own ALC member, you must allocate at least all the above files.

The DATEREC file is ignored when you use CTMAPI to order jobs.

#### **Order or Force Return Codes**

Table 262 shows the return codes that can be returned to the caller (in register R15).

Table 262 Order or Force Return Codes

Return Code	Explanation		
0	The operation was successfully performed.		
4	At least one job was not ordered, due to one of the following:		
	<ul> <li>missing calendar</li> <li>a problem was encountered in the scheduling table</li> <li>a PREV or NEXT date condition was missing</li> <li>CTMX001 cancelled the order</li> </ul>		
8	An error occurred, and the order stopped while being processed.		
12	Syntax error in the command.		
16 or more	Severe error in CTMAPI. The order stopped while being processed.		

#### **Order or Force Performance Considerations**

There are no specific performance considerations with regard to the Order itself. However, using an IF statement can affect the overall performance. For information regarding the impact of an IF statement on performance, see "Performance Considerations for Selection Criteria" on page 880.

## **Order or Force Security Considerations**

The exit called during the Order process is CTMSE01. The files that are accessed, and the type of access, are summarized in Table 263.

Table 263 Files Accessed during the Order or Force Process

File Name	Type of Access	
Active Jobs File	Read and Write	
Calendar	Read	
Statistics File	Read	
Unit Definition	Read	

# 2. Create, Order, or Force New Tables

CTMAPI enables the user to create job scheduling tables, then order or force those tables. You can order or force a job scheduling table to the Active Jobs File (AJF) even if that table is not in a scheduling library.

This service is similar to that provided by the CTMBLT utility. It is activated by means of appropriate CTMBLT input control statements.

## **Invoking Create, Order or Force New Tables Using Program**

The CTMBLT control statements can be specified in a sequential file pointed to by the DAAPI DD statement, or in an in-core table containing the control statements as 80-byte card images. The first control statement must be the character string 'BLT', beginning in column 1, to indicate that the statements that follow are input for CTMBLT. The rest of the control statements must conform to the usual CTMBLT syntax.

For a full description of the CTMBLT parameters and how to specify whether the scheduling tables should be optionally ordered or forced, see the description of the CTMBLT utility in the *INCONTROL* for *z/OS Utilities Guide*.

The SYSPRINT DD statement must be pre-allocated to the step. The output of CTMAPI is written to this file. A return code is returned in register R15. A full list of valid return codes is provided in "Order or Force Return Codes" on page 867. When M is specified in the CTMBLT control parameter OPTION, appropriate messages (BLT89AI) are issued to the job log for each scheduling table that is created.

#### **Create, Order or Force Allocations**

The default ALC member used by the CTMBLT service is ALCMBLT. This allocates the CONTROL-M AJF, IOA LOG, IOA calendars, CONTROL-M Statistics file, and UNITDEF member. These files are required only if the user requests the ordering or forcing of the scheduling tables that are built.

#### **Create, Order or Force Return Codes**

Table 264 shows the return codes that can be returned to the caller (in register R15).

Table 264 Create and/or Order or Force New Tables Return Codes

Return Code	Explanation		
0	The operation was successfully performed.		
8	n error occurred. The table was not built, or not ordered.		
12	Syntax error in the command.		
16 or more	Severe error in CTMAPI.		

## **Create, Order or Force Performance Considerations**

There are no specific performance considerations with regard to CTMBLT itself.

### **Create, Order or Force Security Considerations**

When using the CTMBLT service to create, order, or force a new table, the security considerations are the same as those described in "Order or Force Security Considerations" on page 868.

Table 265 shows the files that are accessed when Order or Force is requested, and the type of access.

Table 265 Files Accessed during the Create, Order or Force Process

File Name	Type of Access	
Active Jobs File	Read and Write	
Calendar File	Read	
Statistics File	Read	
Unit Definition File	Read	

# 3. AJF Actions

Using this type of call to CTMAPI, various actions can be performed against jobs residing in the Active Jobs file (AJF). This service can be called from any environment, with few differences between environments. The full syntax for this service is as follows:

```
AJF{HOLD|FREE|DELETE|UNDELETE|CONFIRM|RERUN|REACT|FORCEOK|KILL}
{selection_criteria} [selection_criteria]...
[IF if_statement]
```

The *selection_criteria* and *if_statement* parts of the command are described in "Conditional Requests and Selection Criteria" on page 879, and the selection criteria are detailed in Table 270 on page 880.

# **AJF Action under Batch, REXX or CLIST**

When called from a batch job or step or from a REXX or CLIST, the AJF statement is specified as one of the following:

a statement in the format

EXEC CTMAPI PARM='AJF variable'

in a sequential file pointed to by a DAAPI DD statement

In this type of call, only a return code is returned in register R15. A full list of valid return codes is provided in "AJF Action Return Codes" on page 871.

If multiple commands are entered in a DAAPI DD statement, the final return code is the highest return code from any of the commands.

### **AJF Action using Program**

When called from a program, the simplest method of requesting the appropriate action against a job is to pass the above statement to CTMAPI as a standard parameter. Alternatively, you can use the conversational mode of the interface, where CTMBAPI area is passed as the parameter, and fields in it identify the request. This mode is described in "Conversational Mode using Program" on page 882.

The following statement is an example of the standard method of executing CTMAPI in batch job and passing it a <code>HOLD</code> request:

//DAAPI DD *
AJF HOLD MEM=DEFSCHD1 JOB=JOBA OID=OAS45 ODATE=090601 IF STATE=FREE

#### - NOTE -



The HOLD command for the job is issued only if its prior STATE was FREE.

#### **AJF Action Allocations**

The default ALC member used by the AJF Action service is ALCMAJF, which dynamically allocates the AJF only. If you choose to prepare your own ALC member, you must ensure that you allocate at least the above file.

#### **AJF Action Return Codes**

Table 266 shows the return codes that can be returned to the caller (in register R15).

Table 266 AJF Action Return Codes

Return Code	Explanation	
0	The operation was successfully performed.	
4	he operation was not performed. The selection criteria or IF statement vere not matched.	
8	The operation could not be performed.	
12	Syntax error in the command.	
16 or higher	evere error in CTMAPI.	

#### **AJF Action Performance Considerations**

There are no specific performance considerations with regard to the Action itself. However, the Selection Criteria or IF statement can significantly affect the overall performance. For information regarding the impact of Selection Criteria and/or IF statements on performance, see "Performance Considerations for Selection Criteria" on page 880.

# **AJF Action Security Considerations**

The exit that is called during execution of the action is CTMSE08.

Table 267 shows the files that are accessed, and the type of access.

**Table 267 Files Accessed during the AJF Action Process** 

File Name	Type of Access	
Active Jobs File	Read and write.	

# 4. Search

The Search function can be used to check the existence of a job in the AJF. It can be called from any environment. However, the AJF entry of the job can only be returned to the caller by using the CTMBAPI mode. Under all other environments, only the return code is returned to the caller, indicating whether or not the job exists in the AJF.

This function should only be used when you want to execute your own process based on the result of this search. If you want to execute one of the other CTMAPI functions based on the Search result, it is recommended that you use the conditional form of that function instead.

The full syntax for the Search call is as follows:

SEARCH selection_criteria [selection_criteria]...

The various valid *selection_criteria* are described in "Conditional Requests and Selection Criteria" on page 879 and Table 270 on page 880.

#### Search under Batch, REXX or CLIST

When called from a batch job or step or from a REXX or CLIST, the Order statement is specified as one of the following:

a statement in the format

EXEC CTMAPI PARM='SEARCH variable

in a sequential file pointed to by DD statement DAAPI

In this type of call, only a return code is returned in register R15. A full list of valid return codes is provided below.

If multiple commands are entered in DAAPI, the final return code is the highest return code from any of the specified commands.

## **Invoking Search from a Program**

When called from a program, the simplest method of searching for a job is to pass the Search call statement to CTMAPI as a standard parameter. Alternatively, you can use the conversational mode of the interface, where the CTMBAPI area is passed as the parameter, and fields in it identify the requested job. This mode is described in "Conversational Mode using Program" on page 882. As mentioned earlier, the advantage of using the CTMBAPI mode is that your program gets back from CTMAPI the entry of the job, mapped by CTMBJSE DSECT, as described in "The Status Reply DSECT (CTMBJSE)" on page 889.

The following statement is an example of the standard method of executing CTMAPI in batch job and passing it a SEARCH request:

//DAAPI DD *
SEARCH MEM=DEFSCHD1 JOB=JOBA OID=OAS45 ODATE=090601

#### **Search Allocations**

The default ALC member used by the Search service is ALCMAJF, which dynamically allocates the AJF only. If you choose to prepare your own ALC member, you must ensure that you allocate at least the above file.

#### **Search Return Codes**

Table 268 shows the return codes that can be returned to the caller (in register R15).

**Table 268 Search Action Return Codes** 

Return Code	Explanation			
0	The job exists.			
4	The job was not found.			
8	The operation could not be performed.			
12	Syntax error in the command.			
16 and higher	Severe error in CTMAPI.			

#### **Search Performance Considerations**

There are no specific performance considerations with regard to the Search itself. However, the Selection Criteria can significantly affect the overall performance. For information regarding the impact of Selection Criteria on performance, see "Performance Considerations for Selection Criteria" on page 880.

## **Search Security Considerations**

This function does not call any security exit during the Search process.

Table 268 shows the files that are accessed, and the type of access.

Table 269 Files Accessed during the AJF Action Process

File Name	Type of Access
Active Jobs File	Read

# 5. Global Variables

You can use CTMAPI to Set and Checkpoint variables in the IOA Variable Database.

■ The SET command readies the global variables to be written to the IOA Variable Database, but does not actually write them there.

- The CKP (CHECKPOINT) command writes all variables which have been SET to the IOA Global Database.
- The SETCKP command combines the action of both the SET and the CKP commands.

When readying many variables to be written, for performance reasons it is recommended that they first be SET, and only afterwards should the CKP command be issued. When only a single variable is to updated, the SETCKP command may be issued, instead of issuing the SET followed by the CKP command.

The resolve option is available only when CTMAPI is called in Conversation mode.

The full syntax for this CTMAPI service is

GLOBAL {SET	T   SETCKE	CHECKP	OINT   CKP}		
$\{PRODUCT=x\}$	APPL=xxx	GROUP = xxx	MEMBER=xxx	$VAR=\%\Xxx$	VALUE=xxx

#### – NOTE –



Separate the parameters in GLOBAL commands with blanks, not commas.

If the action to be performed is SET or SETCKP, the name of the variable must be supplied. The keyword parameters are used to define the variable name.

For more information on the actions and components of the variable name, see the IOA administration chapter in the *INCONTROL* for *z/OS* Administrator Guide.

#### **Global Variables under Batch REXX or CLIST**

If you are calling this function from a batch job, REXX, or CLIST, the GLOBAL statement can be specified in one of the following:

a statement with the following syntax

EXEC CTMAPI PARM='GLOBAL action | varname'

#### where:

- *action* has one of the following values:
  - SET
  - SETCKP
  - CHECKPOINT
  - CKP
- varname is the name of a global variable
- a sequential file pointed to by the DAAPI DD statement

If you use a DAAPI file, you can insert multiple commands.

# 6. Conditions and Resources

You can use CTMAPI to add, delete, change, or query for conditions or quantitative and control resources. In this case, CTMAPI actually calls the IOACND utility. The syntax is the same as the syntax for the IOACND utility, prefixed with the command word COND or RES.

```
COND {ADD | DELETE | CHECK} COND cond-name ODATE
RES {ADD | DELETE} CONTROL control-name {EXCLUSIVE | SHARE}
RES ADD RESOURCE res-name quantity
RES DELETE RESOURCE res-name
RES CHANGE RESOURCE res-name [-|+]quantity
```

For more information on the syntax of the IOACND utility and its characteristics, see the IOACND section in the *INCONTROL* for *z/OS Utilities Guide*.

# 7. CTMAS Active Job Download Filtering

You can use the CTMAPI EMDOWNLD service to control which job in the Active Job File will be downloaded to CONTROL-M/EM during download processing of the CONTROL-M Application Server (CTMAS).

- The EXCLUDE command excludes the job(s) specified from being downloaded to CONTROL-M/EM
- The ACCEPT command indicates to CTMAPI to no longer exclude the job(s) specified from being downloaded to CONTROL-M/EM

The full syntax for this service is as follows:

```
EMDOWNLD{EXCLUDE|ACCEPT}
{selection_criteria} [selection_criteria]...
[IF if_statement]
```

The *selection_criteria* and *if_statement* parts of the command are described in "Conditional Requests and Selection Criteria" on page 879. The selection criteria are detailed in Table 270 on page 880.

#### − NOTE



If the EXCLUDE or ACCEPT action is requested for a group entity, the action will be performed on all jobs belonging to the group.

The EMDOWNLD service also includes the following specific syntax commands:

- EXCLUDE LIST List all jobs currently excluded from download to CONTROL-M/EM
- ACCEPT ALL Include all currently excluded jobs in the next download to CONTROL-M/EM

#### **EMDOWNLD Action under Batch, REXX or CLIST**

The EMDOWNLD service can be called from a batch job or step or from a REXX or CLIST, in one of the following ways:

a statement in the format

#### EXEC CTMAPI PARM='EMDOWNLD variable'

in a sequential file pointed to by DD statement DAAPI

In this type of call, only a return code is returned in register R15. A full list of valid return codes is provided in "AJF Action Return Codes" on page 871.

If multiple commands are entered in DAAPI, the final return code is the highest return code from any of the specified commands.

#### **EMDOWNLD Action Allocations**

For information on EMDOWNLD action allocations, see "AJF Action Allocations" on page 871.

#### **EMDOWNLD Action Return Codes**

For information on EMDOWNLD action return codes, see "AJF Action Return Codes" on page 871.

#### **EMDOWNLD Action Performance Considerations**

For information on EMDOWNLD action performance considerations, see "AJF Action Performance Considerations" on page 872.

## **EMDOWNLD Action Security Considerations**

For information on EMDOWNLD action performance considerations, see "AJF Action Performance Considerations" on page 872.

## **Examples**

## **Example 1**

Prevent download of job: Orderid 000DB

//S1 EXEC PGM=CTMAPI, PARM='EMDOWNLD EXCLUDE OID=000DB'

#### **Example 2**

Allow download of job: Member name BR14

//S1 EXEC PGM=CTMAPI, PARM='EMDOWNLD ACCEPT MEMBER=BR14'

#### **Example 3**

List all jobs currently excluded from download to CONTROL-M/EM

```
//S1 EXEC PGM=CTMAPI, PARM='EMDOWNLD EXCLUDE LIST'
```

# **Conditional Requests and Selection Criteria**

Many services can be conditionally executed based on various terms and conditions. This topic describes in more detail the various criteria that can be used.

Poor usage of selection criteria can dramatically impact the overall performance. Before using such selection criteria, read "Performance Considerations for Selection Criteria" on page 880.

Character fields marked with an * (Asterisk) to the right of the field are used as a prefix for the specified selection criteria. No masking is allowed in any other field.

For example, if you specify MEM=ABC*, all jobs with the MEMNAME prefix "ABC" will be selected.

The full syntax for the selection criteria is as follows:

```
IF {[MEM=memname*] |
                     [GROUP|GRP=group_name*]
 [JOB=job_name*] | [JOBID=jes_job_number]
 [OWNER=owner*] | [OID=orderid] | [ODATE={ODAT|date}] |
 [STATUS={WAIT_SCH
          WAIT_CONF
          WAIT_PIPE
          WAIT_ORD
          EXEC_ERR
          EXEC WSUB
          EXEC_INQ
          EXEC_NJE
          END_OK
          END_OK_FOK
          END_NOK_ABND
          END_NOK_JCLE
          END_NOK_UNKW
          END_NOK_CC
          END_NOK_NSUB
          END_NOK_DISA
          EXIST
          NOTEXIST
          NOT_DELETED
          DEL ]
    [STATE={HOLD|FREE}]}
  [{AND|OR} selection2 }]
```

Table 270 shows the meanings of the parameters in that statement.

**Table 270 Selection Criteria Parameters** 

Parameter	Description			
MEM	Member name of the job			
GROUP (or GRP)	Group name of the job			
JOB	Job name of the job (valid only after the job was submitted)			
JOBID	JES job number (valid only after the job was submitted)			
OWNER	Owner of the job			
OID	The CONTROL-M Order ID of the job			
ODATE	The ODATE of the job. Valid values are:  ■ ODATE – The current CONTROL-M ODATE  ■ DATE - Same as ODATE  ■ date – The full date, in the format YYMMDD, MMDDYY, or DDMMYY, depending on your site format			
STATUS	For an explanation of these statuses, see Table 276 on page 890.			
STATE	Whether the job is held or free. Valid values are:  ■ HOLD – The job is held. ■ FREE – The job is free.			
selection2	Any of the above parameters.			

Multiple IF statements can be specified, connected to each other using regular Boolean logic, including expressions inside parentheses.

## **Performance Considerations for Selection Criteria**

The overall performance of each call to CTMAPI is largely dependent on the selection criteria. These must be carefully considered.

An important factor affecting overall performance is the uniqueness of the selection criteria. If very few jobs in the Active Jobs file conform to your selection criteria, then very few job records will have to be handled. For example, if you search for a specific Order ID, the result will be the reading of only a few index records and only one job record. On the other hand, if you search for all jobs with a member name starting with ABC, the API must read many job records as well as the index records.

You can greatly improve overall performance by using indexed fields in the selection criteria. This results in a faster and more efficient search. The use of non-indexed fields causes a sequential search through the Active Jobs file, which is very slow and inefficient.

Table 271 shows the attributes of each selection criteria parameter.

**Table 271 Selection Criteria Parameter Attributes** 

Parameter	Indexed	Unique	Notes
MEM	Yes	No	
GROUP	Yes	No	
JOB	Yes	No	Valid only after job submission
JOBID	No	No	Valid only after job submission
OWNER	Yes	No	
OID	Yes	Yes	
ODATE	No	No	
STATUS	Yes	No	EXIST and NONEXIST statuses are not indexed.
STATE	Yes	No	

As Table 271 shows, OID is the best choice for selection criteria, since it is both indexed and unique. On the other hand, ODATE and JOBID are the worst choices for selection criteria, since they are neither indexed nor unique. If you must use one of the non-indexed search criteria, BMC Software recommends using it in a combination with other indexed criteria.

Another factor affecting overall performance is the complexity of any AND or OR statements that qualify the selection criteria. Statements included in an AND or OR section of the selection criteria are each handled separately, one by one, as if each is a fully qualified selection criteria, and the whole Boolean sentence is verified only after each such statement is checked.

# **Search Security Considerations**

This function does not call any security exit during the Search process.

Table 272 shows the files that are accessed, and the type of access.

Table 272 Files Accessed during the AJF Action Process

File Name	Type of Access	
Active Jobs File	Read	

# **Return Codes**

CTMAPI return codes are returned in register R15. They are also returned in the following fields:

- BAPIRC
- BAPIRSN
- BAPIURC

The following are the types of failure of CTMAPI:

■ CTMAPI itself encountered a problem that prevented it from calling a service.

In this case

- register R15 has a value higher than 8
- the reason code is returned in the BAPIRSN field
- The service was activated, but failed to perform the desired action.

In this case

- register R15 has a value of 8
- the return code from the service is returned in the BAPIURC field

# **Conversational Mode using Program**

This type of call is intended for use by programs. It enables the program to pass requests, accept replies, respond on the basis of the reply, and maintain communication between the program and the API.

The basic communication area, which is passed back and forth between the calling program and the API, is mapped by the Assembler DSECT CTMBAPI, which can be found in the IOA MAC library. Different fields in this DSECT identify the request, specify the selection criteria, and pass the address of the area in which replies to the caller are to be returned.

# **Input and Output Registers**

On input to any CTMAPI service, the contents of the general-purpose registers should be as shown in Table 273.

Table 273 Contents of Registers on Input to CTMAPI

Register	Contents
R0	Irrelevant
R1	Address of parameter list, where the first (and only) parameter has its high-order bit turned on (the VL parameter of the CALL macro). This parameter points to a 2-byte field containing the length of the CTMAPI DSECT. The 2-byte length field must be followed by the double word aligned CTMAPI DSECT.
R2 through R12	Irrelevant
R13	SAVE AREA address of caller
R14	Return address
R15	CTMAPI entry point address

On return, all registers are restored by CTMAPI, and a return code is returned in register R15. In this appendix, the return codes and their meanings are explained under each service.

# **Example – Invoking CTMAPI in BAPI mode**

The following example may be used to invoke CTMAPI in BAPI mode:

```
LOAD EP=CTMAPI
              R15,R0
        0.1
              BAPIADR, X'80'
        ΙΑ
              R1, BAPIADR POINT TO BAPI AREA ADDRESS
        BALR R14,R15
                           CALL CTMAPI
              A(BAPIDSLN)
BAPIADR DC
                            BAPI DSECT + HALFW LEN
        DS
              0 D
                           DOUBLEWORD ALIGNMENT
        DS
              CL6
                            6 BYTE FILLER
             Y(BAPILEN)
BAPIDSLN DC
                           HALFWORD LENGTH FROM CTMBAPI DSECT
        CTMBAPI DSECT=NO
```

# CTMBAPI DSECT

This section describes in more detail

- how to use the CTMBAPI DSECT
- what fields the caller should set
- what fields are used to return the result

The explanation assumes the use of Assembler language. However, you can use other high level languages to implement most of the services, provided the language you use conforms to the standard calling conventions in Table 273.

The type of service is identified in one or both of the following ways

- as a command within a buffer The start address of the buffer is passed to the API using the BAPICMDA field. The command length is passed using the BAPICMDL field.
- using the BAPICMD field, which identifies the type of service

If both are specified, the command takes precedence.

CTMBAPI is composed of

- a fixed part
   This is used to identify the requested service, together with other necessary fields.
- a variable (extension) part
   This is in variable format, where each service uses a different extension.

For each service, the format of each extension is documented in the following sections of this appendix, for example in "Status Extension" on page 887, "Order Extension" on page 892, "AJF Action Extension" on page 895, and so on.

The fields in the fixed (header) part are summarized in Table 274. The values in the columns in Table 274 have the following significance:

- In the column headed "Optional or Mandatory"
  - M means mandatory
  - O means Optional
- In the column headed "In or Out"
  - I means Input
  - O means Output

- In the column headed "Type"
  - CL*nn* means a character field *nn* characters in length, padded with blanks to the right.
    - If omitted, it must be set to blanks.
  - an * (Asterisk) to the right of the CL*nn* entry means that the characters in the field are used as a prefix for the specified selection criteria. For example, if you specify MEMNAME ABC*, all jobs with a MEMNAME prefix of "ABC" will be selected.
  - A means Address, a 4-byte fullword field pointing to an area. If omitted, it must be set to binary zero.
  - F means Fullword, a 4-byte fullword field containing a binary number. If omitted, it must be set to binary zero.
  - H means Halfword, a 2-byte halfword field containing a binary number. If omitted, it must be set to binary zero.
  - Flag means a Flag Byte, where each bit has a separate significance.

Table 274 Fixed Part Values (part 1 of 3)

Field Name	Optional or Mandatory	In or Out	Туре	Usage
BAPICMD	О	Ι	CL1	One byte identifier of the requested service
				Note: If the BAPICMDA field is set to a value other than zero, it takes precedence over the BAPICMD field.
BAPICMDA	0	I	A	Address of command buffer. The command syntax should be identical with the syntax of the individual CTMAPI functions described in this appendix. If set to zero, the requested service must be specified in the BAPICMD field.
BAPICMDL	0	I	F	Length of the command in the command buffer. Ignored if the value in BAPICMDA is zero.

Table 274 Fixed Part Values (part 2 of 3)

Field Name	Optional or Mandatory	In or Out	Туре	Usage
BAPIFLG1	0	I	Flag	General purpose flag.  BAPIF1CN (X'80') – Do not release the working area on return.  Applicable for programs that call the API several times. It is the responsibility of the caller to call the API with the function BAPI_TERM to allow the API to free storage, close files, and so on. Failure to do so may cause unpredictable results, such as
BAPIMCT	M	I or O	A	storage accumulation.  Address of IOA MCT used by the API. The caller must set this field to zero on the first call, and leave it untouched between multiple calls.
BAPIRC	Not applicable	О	Н	Return code returned to the caller. Identical with register R15.
BAPIRPL#	0	0	F	Number of reply slots returned by the API. The size of each slot depends on the service requested.
BAPIRPLC	0	О	F	Address of the first free byte in the reply area. Serves to indicate the end of that area.
BAPIRPLE	O	I	A	End address of the reply buffer.  If BAPIRPLS (in this Table) is set to zero, this field is ignored.  This field informs API of the size of the reply buffer. In some services, such as STATUS, if the reply buffer space is exhausted, a special return code indicating this is returned to the caller. The caller can then again call the API to obtain the rest of the reply.
BAPIRPLS	O	I	A	<ul> <li>Start address of the reply buffer.</li> <li>If set to zero, no reply is returned.</li> <li>If set to a value other than zero, the API returns its replies into this buffer.</li> <li>The reply that the API can return is explained in relation to each service.</li> </ul>

Table 274 Fixed Part Values (part 3 of 3)

Field Name	Optional or Mandatory	In or Out	Туре	Usage
BAPIRSN	Not applicable	0	Н	Reason code returned to the caller.  Valid reason codes are documented internally in the CTMBAPI macro.
BAPISIGN	M	I	CL4	DSECT eye-catcher 'BAPI'
BAPIURC	Not applicable	О	Н	Return code returned to CTMAPI from the invoked utility if that utility failed. This return code is set only if CTMAPI ended with return code 8. Otherwise, the CTMAPI return code itself identifies the problem.
BAPIVERS	M	I	CL1	DSECT Version. The current version is 1.
BAPIWORK	M	I or O	A	Address of the API work area. This field is used by the API to hold information between calls when more replies must be returned. The caller must set this field to zero and leave it untouched between multiple calls.

# **Status Extension**

The value of 2 (BAPI_M_STATUS) should be set in the BAPICMD field for the Status function.

The Status function can be used to retrieve information about jobs in the Active Jobs file. This service can be called from any environment, but only by using the CTMBAPI mode, that is, using a program, and not by means of a batch statement, REXX or CLIST.

On return, the status of and other information about the job is returned to the caller.

If you only requested one job, for example, Status using OID, the result is returned in the Status extension itself. If more than one job may conform to the selection criteria, for example, the status of MEMNAME ABC, a reply buffer must be supplied into which the API can return a result for each and every job that conforms. If no such buffer is supplied, no reply other than an appropriate return code is returned.

The Status extension fields are summarized in Table 275. If the caller filled in a field, it is used as Search argument, and only jobs that conform to that field are returned. On return from the API, if no reply area has been supplied, and if only one job conforms to the selection criteria, the API will fill in all these fields with actual information about this job. For example, if you specify ABC in BAPISMEM, and there is only one job in the AJF with a matching MEMNAME, such as ABCXYZ, on return from the API this field will hold the value ABCXYZ.

The values in the columns in Table 275 have the same significance as those in Table 274.

Table 275 Status Extension Fields (part 1 of 2)

		In or		
Field Name	Optional	Out	Туре	Usage
BAPISGRN	O	I or O	CL20*	Group name.
BAPISHLD	0	I or O	CL1	Hold state. Valid values are:
				■ H (Hold)
				■ F (Free)
BAPISJID	O	I or O	CL5	JES job ID (job number).
BAPISJNM	О	I or O	CL8*	Job name. Valid only after job submission.
BAPISLIB	О	I or O	CL44*	Scheduling library from which the job was ordered.
BAPISMEM	O	I or O	CL8*	MEMNAME.
BAPISODT	О	I or O	CL6	ODATE of the job. This must be fully specified. Prefixing is not supported.
BAPISOID	O	I or O	CL5	Order ID. If a value is entered, it must include all five characters of the Order ID.
				Prefixing is not supported.
BAPISOWN	O	I or O	CL8*	Owner of the job.
BAPISRBA	Not applicable	О	CL6	RBA of the job, in hexadecimal format.
BAPISRBB	Not applicable	О	CL3	RBA of the job, in binary format.
BAPISRNM	Not applicable	О	Н	Run number of the job.

Table 275 Status Extension Fields (part 2 of 2)

Field Name	Optional	In or Out	Туре	Usage
BAPISSTT	O	I or O	CL15*	Status of job. For a list of valid values, see Table 276 on page 890.  The masking character "*" can be used in any status value which includes an underscore character "_". However, the "*" must follow immediately after the "_".
BAPISTAB	0	I or O	CL8*	Scheduling table from which the job was ordered.
BAPISTAG	О	I	CL20	(Only for jobs that are part of a group) The name of the scheduling tag that made the job eligible for execution.
BAPISTYP	O	I or O	CL3	Task type. Valid values are:  JOB GRP STC CYC EMR CST ECJ EST ECS WRN  Except for GRP, each of these is explained in "TASKTYPE: General Job Parameter" on page 654. GRP is explained in Table 54 on page 171.

## The Status Reply DSECT (CTMBJSE)

The DSECT that formats reply area entries is CTMBJSE. Each entry is 240 bytes long. For REXX parsing, fields in this DSECT are separated by a blank.

You must always allocate an area of 12,000 bytes and code its address in the BAPIRPLS field.

The search criteria can fit multiple jobs on the AJF, up to a maximum of 50 jobs. For example, if you want to process 25 jobs, prepare an area of 12,000 bytes and code its address in the BAPIRPLS field. After returning from the API, the area will contain the details of the 25 jobs. Each job line is detailed in the CTMBJSE DSECT and contains relevant information about the located job.

The number of lines is returned in the BAPIRPL# field. When this field points to the maximum, 50, it is possible that there are more lines that can be returned. In that case, the value of the Utility Return Code field BAPIURC will be 4, and the Reason Code field BAPIRSN will have the value "BAPI_HAVE_MORE_LINES." In such a case, the user program can set bit BAPISPF8 in byte BAPISF1 and call CTMAPI again. This call will retrieve the next 50 lines of output that match the search criteria. When multiple lines are returned, the lines are in the order from the end (the most recent job) to the beginning. There is an option for the calling program to receive only one line of output, by specifying as the value in the BAPISF1 flag byte either BAPIS1ST (first line) or BAPISLST (last line).

Except for field JSESTAT, the meanings of the fields are as described (internally) in the macro CTMBJSE, which is in the IOA MAC library. The JSESTAT field returns the status of the job in the AJF. The CTMAPI status function does not return all the statuses detailed in the *CONTROL-M for z/OS User Guide*. A list of the statuses that can be returned appears in Table 276.

Table 276 Statuses Returnable under the Status Function

Status	Description
DEL	The job was deleted.
END_NOK_ABND	The job ended NOTOK because of an abend.
END_NOK_CC	The job ended NOTOK because of the Condition Code of the job.
END_NOK_DISA	The job ended NOTOK. It disappeared.
END_NOK_JCLE	The job ended NOTOK because of a JCL error.
END_NOK_NSUB	The job ended NOTOK. It was not submitted by JES.
END_NOK_UNKW	The job ended NOTOK for an unknown reason.
END_OK	The job ended OK.
END_OK_FOK	The job was ForcedOK.
EXEC	Job is executing.
EXEC_ERR	Relevant only to group entities. Several of the jobs in the group are still executing, but at least one of them has ended NOTOK.
EXEC_INQ	The job was submitted to JES, but is not yet processing.
EXEC_NJE	The job is being executed at a remote NJE node.
EXEC_WSUB	Wait submission. The job was selected, but it is still waiting for CONTROL-M to submit it to JES.
NOT_DELETED	The job was not deleted.
WAIT_CONF	Wait for confirmation.
WAIT_ORD	The ordering of a group is not yet complete. The group is still in the order process.
WAIT_PIPE	Waiting for all members of the pipe to be ready for submission.
WAIT_SCH	Wait Schedule.
EXIST	The job exists on the Active Jobs file.
NOTEXIST	The job does not exist on the Active Jobs file.

#### **Status Allocations**

The ALC member used by the Status service as the default is ALCMAJF, which dynamically allocates the AJF only. If you choose to prepare your own ALC member, you must ensure that you allocate at least the above file.

#### **Status Return Codes**

Table 277 shows the return codes that can be returned to the caller (in register R15).

Table 277 Status Return Codes

Return Code	Explanation
0	The operation was completed OK.
	If a reply buffer was supplied, but was exhausted, meaning that not all the statuses could be returned into the supplied buffer, a special reason code, 286, is returned in the BAPIRSN field to indicate that there are more replies.
4	The job was not found.
8	The operation could not be performed.
12	Syntax error in the command.
16 and higher	Severe error in CTMAPI.

#### **Status Performance Considerations**

The Status function searches the AJF for the requested jobs. More than one job may conform to the selection criteria specified in the CTMBAPI DSECT. In that case, a Job Status Element (JSE) entry is returned to the caller for each job.

The Selection Criteria can significantly affect overall performance.

- The more specific the request, the fewer the jobs that must be read and returned to the caller. For example, if you request status information for all jobs starting with the letter A, the function must read a large part of the AJF, degrading its performance.
- Pay special attention to whether your search criteria are indexed. If you ask for status information about jobs with a selection criteria that is not indexed, for example, from a specific ODATE, without any indexed selection criteria, the whole AJF must probably be read.

The impact of Selection Criteria on overall performance is described in "Performance Considerations for Selection Criteria" on page 880.

# **Status Security Considerations**

This function does not call any security exit during the Status process.

The files that are accessed, and the type of access are summarized in the following table:

**Table 278 Files Accessed during the AJF Action Process** 

File Name	Type of Access
Active Jobs File	Read

# **Order Extension**

The value that should be set in the BAPICMD field for this function is 1 (BAPI_M_ORDER).

You can use the Order function to order jobs to the AJF. You can call this function from any environment, but only by using the CTMBAPI mode. The function uses the usual CONTROL-M order process to put the requested job on the AJF. The return code will appear in the BAPIURC (Utility Return Code) field.

If CTMAPI fails to invoke the order process, register R15 will contain a value of 8 or higher, and the reason code will appear in the BAPIRSN field.

You can request a detailed reply from the order process, using the following procedure:

- 1 Prepare a memory area.
- **2** Pass the start address in the BAPIRPLS field.
- **3** Pass the address of the last byte of this area in the BAPIRPLE field.

After returning from the order process, the BAPIRPLC field will point to the last byte replied. The BAPIRPL# field will contain the number of reply lines. For each job processed by the order process, a reply line will be returned detailing the job identifiers and the RC of the order for this specific job. This is in contrast to the usual output lines of the order process that are issued only for jobs that have actually been ordered. Details of the reply line are specified in the CTMBAPO DSECT.

Table 279 contains a summary of the CTMBAPI Order input fields. You must ensure that all fields whose type is CL are initialized with BLANKS, and those with type X are initialized to binary zeros.

Table 279 Order Fields (part 1 of 2)

Field Name	Optional	In or Out	Туре	Usage
BAPIODSN	N	I	CL44	Scheduling table DS name. Mutually exclusive with BAPIODD.
BAPIOMEM	N	I	CL8	Member name
BAPIODD	О	I	CL8	Scheduling table DD name. Mutually exclusive with BAPIODSN.
BAPIOJOB	О	I	CL8	Job name. Enter ''(Blank) to order all jobs in the table.
BAPIODAT	O	I	CL6	ODATE. Default is the current Odate.
BAPIORBA	0	I	CL6	RBA of the group entity when a Dynamic Group Insert is performed.
BAPIOF1	O	I	XL1	<ul> <li>Flags byte. Valid values are:</li> <li>■ X'80' - Force the table</li> <li>■ X'40' - Insert the job into a group entity that already exists on the AJF</li> <li>■ X'20' - Allow duplicate jobs in the group when dynamically inserting a job into the group</li> <li>■ From X'10' through X'01' - These bits are reserved for internal use</li> </ul>

Table 279 Order Fields (part 2 of 2)

Field Name	Optional	In or Out	Туре	Usage
BAPITAG#	О	Ι	XL2	Number of tag statements that follow this field.
				This field is for users who want to implement the IGNORE and/or SELECT TAG logic that is discussed on connection with the utility CTMJOB in the CONTROL-M chapter of the INCONTROL for z/OS Utilities Guide.
				After this field, you should code the matching number of BAPITGNM statements that define the tags themselves.
BAPITGIN	О	I	CL1	Ignore or Select indicator. Valid values are:  ■ I – Ignore ■ S – Select
BAPITGNM	0	I	CL20	Tag name

# **Order Return Codes**

Table 280 shows the return codes that can be returned to the caller (in register R15).

**Table 280 Order Return Codes** 

Return Code	Explanation
0	Order completed OK. If a reply buffer is specified in the BAPIRPLS field, a reply line is returned for each job.
4	Order completed OK, but the order process issued a warning. This usually occurs when one of the specified calendars was not found.
8	The operation could not be performed. The Order process encountered a severe error.
12	Syntax error.
16	Severe error in CTMAPI, such as a failure to get memory or a failure to open a file.

## **Order Reply**

The conversational mode (BAPI) order process can return a reply line for each job processed. The reply line is mapped by DSECT CTMBAPO, which is described in more detail in "CTMBAPO" on page 902.

#### **Order or Force Allocations**

For full information, see "1. Order or Force Existing Jobs" on page 865.

## **Order or Force Security Consideration**

For full information, see "1. Order or Force Existing Jobs" on page 865.

# **AJF Action Extension**

The AJF Action function can be used to perform basic Active Environment screen (Screen 3) actions upon jobs in the AJF. Using the BAPI interface, a user program is able to perform actions such as holding, freeing, deleting jobs in the AJF in much the same manner as the user can from Screen 3.

For this function, set the value in the BAPICMD field to 3 (BAPI_M_ACT).

The Action function can be called from any environment. The input contains the following types of input:

- to identify the job
- to define the Action upon the job

These are described in the following sections.

# **Identifying the Job**

The first type of input identifies the job, using the field names in Table 281.

**Table 281 AJF Action Parameters** 

Field Name	Optional	In or Out	Туре	Usage
BAPIAMEM	O	I	CL8	Member name in table.
BAPIAJNM	O	I	CL8	Job name, in JCL.
BAPIAOWN	O	I	CL8	Owner ID of the job.
BAPIAJID	O	I	CL5	JOBID as returned by JES.
BAPIAOID	O	I	CL5	CONTROL-M ORDERID.
BAPIAGRN	O	I	CL20	Group name.
BAPIRBAN	O	I	XL3	RBA in binary format.
BAPIRBAC	О	I	CL6	RBA of the job in characters, with each character representing a hexadecimal digit.

CTMAPI uses this variable to find the job in the same way it does a search. For information concerning performance and security, see "Create, Order or Force Performance Considerations" on page 869.

# **Defining the Action**

To define the Action, you must set a 1-byte field called BAPIAACT with one of the values in Table 282

Table 282 AJF Action BAPIAACT Field Values

Value	Explanation
BAPIAHLD	Hold
BAPIAFRE	Free
BAPIADEL	Delete
BAPIAKIL	Kill
BAPIAUND	Undelete
BAPIARER	Rerun
BAPIARCT	React
BAPIAFOK	Force OK
BAPIACON	Confirm

#### **Action Return Codes**

The CTMAPI Action return code is returned in register R15. There are basically two types of failure:

- The CTMAPI program itself encountered a problem which prevented it from calling the service. In this case
  - register R15 has a value higher than 8
  - the reason code is returned in the BAPIRSN field
- The service was activated but failed to perform the desired action. In this case
  - register R15 has a value of 8
  - the return code from the service is returned in the BAPIURC field

Table 283 shows in more detail the return codes that can be returned to the caller (in register R15).

Table 283 CTMAPI Action Return Codes

Return Code	Explanation
0	The action was successfully performed.
8	The action was not successfully performed.
	The field BAPIURC contains a return code indicating the fault.
12	Syntax error.
16	Severe error, such as failure to get memory, or failure to open a file.

#### **Action AJF Allocations**

For information on AJF Actions, see "3. AJF Actions" on page 870.

# **Action Security Considerations**

For information on AJF Action security considerations, see "3. AJF Actions" on page 870.

# **Global Variable Extension**

The Global Variable Extension is used to resolve, set, or checkpoint variables in the IOA Variable Database. For more information on this facility, see the IOA administration chapter in the *INCONTROL* for *z/OS* Administrator Guide.

The value for this function in the BAPICMD field is 6. For more information on the BAPICMD field, see "BAPICMD" on page 885.

Table 284 contains a summary of the CTMBAPI Global Variable Extension input fields. You must ensure that all fields whose type is CL are initialized with BLANKS, and those with type X are initialized to binary zeros.

**Table 284 Global Variable Fields** 

Field Name	Optional	In or Out	Туре	Usage
BAPIGOPT	N	I	XL1	Option byte.
				Valid values are:  ■ X'00' – Resolve Obtain the value of a variable from the database.  ■ X'80' – Set
				Set the value of a variable from the database.
				■ X'40' – Checkpoint Force all changed variables to be written to the database.
BAPIGIOA	0	I	CL8	QNAME
				Default: MCTQNAME
BAPIGAPL	0	I	CL20	Application name.
				Default: NO_APPL
BAPIGGRP	0	I	CL20	Group name.
				Default: NO_GROUP
BAPIGMEM	0	I	CL20	Member name.
				Default: NO_MEM
BAPIGVAR	N	I	CL256	Variable name.
BAPIGVAL	N	I/O	CL256	Variable value.
BAPIGPRC	0	I	CL1	INCONTROL product.
				Default: 'M'

#### **Global Variable Return Codes**

Table 285 shows in more detail the Global Variable return codes that can be returned to the caller (in register R15).

Table 285 Global Variable Return Codes

Return Code	Explanation
0	The action was successfully performed.
8	The action was not successfully performed.
	The field BAPIURC contains a return code indicating the fault.
12	Syntax error.
16	Severe error, such as failure to get memory, or failure to open a file.

# **Quantitative Resource Extension**

The Quantitative Resource Extension function is used to query the status of a quantitative resource in the CONTROL-M Resources file. It can be called from any environment by means of the CTMBAPI mode.

Use BAPI_M_RES to set the value for this function in the BAPICMD field to 4.

**Table 286 CTMAPI Quantitative Resource Fields** 

Field Name	Optional	In or Out	Туре	Usage
BAPIRESN	N	I	CL20	Name of the resource. This serves as the sole key to the search.
BAPIRESX		0	XL2	Maximum quantity defined for this resource.
BAPIRESQ		0	XL2	Quantity currently held by jobs in the AJF.
BAPIRESP		О	XL1	If the resource is reserved for a critical path job, this field will contain the priority of this job, which will be from 1 through 9.

#### **Quantitative Resource Return Codes**

The result is returned directly to the BAPI DSECT as specified below.

**Table 287 CTMAPI Quantitative Resource Return Codes** 

Return Code	Explanation
0	The operation completed OK. The output fields in the BAPI DSECT are updated.
4	The resource was not found in the file.
16	Severe error encountered, such as failure to get memory or error in accessing the file.

## **Quantitative Resource Security Considerations**

The security exit called is IOASE07.

### **Quantitative Resource Allocations**

The files that are accessed, and the type of access, are summarized in Table 288.

**Table 288 CTMAPI Quantitative Resource File Allocation** 

File Name	Type of Access
RES	Read

# **Create and/or Order or Force a Table (BLT)**

The BLT function invokes the CTMBLT utility to create, save, and optionally order a table on the fly.

Unlike the other functions implemented through BAPI extension, this feature does not contain a separate extension where you define the input parameters. Instead

- 1 Set the BAPICMD field to the value BAPI_M_BLT.
- 2 Prepare the input to the API in memory as a regular CTMBLT input stream, as described in the *INCONTROL for z/OS Utilities Guide*, pointed to by the CTMCMDA field.

- **3** Set the length of the input, in bytes, in the BAPICMDL field. Each control input statement must be an 80-byte card image.
- **4** Set the reply fields.

When requesting reply fields in this function, through the BAPI interface, you receive reply lines from both the CTMBLT function and CTMJOB. For more information on the reply input and output fields, see "CTMBAPO" on page 902.

### **BLT Action Return Codes**

Table 289 BLT Action Return Codes

Return Code	Explanation	
0	The action was successfully performed.	
8	The utility did not perform the action.	
	The field BAPIURC contains a return code indicating the fault.	
12	Syntax error.	

## **BLT Reply**

The conversational mode (BAPI) BLT function can return a reply line for

- each Table that was saved
- each job that was processed

The reply line is mapped by the CTMBAPO DSECT, which is described in more detail in "CTMBAPO" on page 902.

## **BLT Security Considerations**

When creating and saving scheduling tables, no IOA security exits are invoked to check the authority of the user to access the scheduling table. If the table must also be ordered, CTMSE01 will be called to verify that the user has the authority to order the table.

### **BLT Resource Allocations**

Table 290 shows the files that are accessed, and the type of access.

**Table 290 CTMAPI Quantitative Resource File Allocation** 

File Name	Type of Access
Active Jobs File	Read and Write
Calendar File	Read
Statistics File	Read
Unit Definition File	Read

# **Replies**

The BAPI feature returns output to the customer in the following ways:

- a return code
- setting output fields in the BAPI DSECT These fields were individually described in "CTMBAPI DSECT" on page 884.
- an output buffer returned by the Status service and described by the CTMBJSE DSECT
- the replies returned by the Order and BLT functions, as described in "CTMBAPO" on page 902

# **CTMBAPO**

When in CTMBAPI mode, CTMAPI serves as an interface between a user program and a CONTROL-M service. Some CTMAPI services have been modified to enable them to return lines of replies into customer-supplied memory to detail their activity. Currently this facility can be provided by

- the BLT process
- the Order process

For example, if the proper instructions are given, the Order process will return a reply line for each job which it processes. This contrasts with normal processing, where a line of output is not written until a job is actually placed on the AJF or a severe error has occurred. You can then act upon and/or process the reply lines.

To use this facility, you must supply the API with the pointers required to trigger the reply mechanism. These are supplied through the calling program. Table 291 shows the pointers and the fields in which they are supplied.

**Table 291 CTMAPI Reply Mechanism Trigger Pointers** 

Field	Information Required
BAPIRPLS	The starting address of the reply buffer.
BAPIRPLE	The address of the last byte in the reply buffer.

When BAPI returns, the BAPIRPLC field will point to the last byte actually written to the reply buffer, and the BAPIRPL# field will contain the number of lines put there. The API ensures that the value in the BAPIRPLC field never exceeds that set by the BAPIRPLE field. Each line added to the reply buffer will start with the current BAPIRPLC and will update it. BMC Software recommends that this field be initialized to zero. If this field is not zero, API treats the value as the starting address of the next reply line. This can be used by an application to accumulate reply lines across several invocations of CTMAPI.

Each line in the buffer is mapped by the CTMBAPO DSECT. Each line starts with a half-word that contains its length (BAPOLEN), and another two bytes that identify the service that produced the reply line (BAPOID). The identification of each reply line is mandatory, since a called service can call other CONTROL-M services which, in turn, will place their reply lines in the buffer. By using the identification of each reply line together with the contents of the BAPIRPLC field and the BAPIRPL# field, you can code a routine to scan and filter reply lines.

## **Date Format Considerations**

The format of all the date fields, both input and output, depends on your site standard. It follows that when you prepare the input to CTMAPI, you must know your site standard.





# CONTROL-M for z/OS Unix System Services (USS)

This Appendix discusses the implementation of CONTROL-M in the IBM OS/390 or z/OS Unix Systems Services (USS) environment.

In this appendix, the term "MVS" includes MVS, S/390, OS/390, and z/OS.

# **Implementation Options**

The use of USS with CONTROL-M for z/OS can be implemented in different ways, depending on your system and the way in which you use it.

Choose one of the following implementation options:

- Use MVS to run USS applications.
- Have MVS support USS in the same manner as it supports other Unix platforms, such as CONTROL-M/Enterprise Manager, CONTROL-M/Server, and CONTROL-M/Agent.
- Integrate the architecture of SAP R/3 running on USS with the MVS platform.

These options are discussed individually in this Appendix.

# **OS/390-Oriented Implementation**

CONTROL-M for z/OS fully supports the USS environment without any need for modifications.

CONTROL-M for z/OS manages all USS batch processes and integrates them with batch activities on

- the local MVS system
- all other platforms across the network

For CONTROL-M to submit and control all USS executions, all that is required is the definition of a single JCL member. This member contains a USS shell activation program that is supported by the MVS operating system. AutoEdit variables are used to define all elements of the USS task, such as the name of the script, the script parameters, the job name and the script location. When CONTROL-M submits the JCL, all the AutoEdit values are resolved and the JCL is submitted with its corresponding values. The JCL then submits the appropriate script under USS. CONTROL-M reads the return code of the script execution from the JCL sysout, and proceeds accordingly.

A sample JCL member is shown in Figure 372.

#### Figure 372 JCL for USS Execution

```
//jobname JOB (account_info),REGION=5000K
//STEPNAME EXEC PGM=BPXBATCH
// PARM='sh /u/usr_id/%%MYSCRPT'
//STDOUT DD PATH='/u/usr_id/stdout.f',PATHOPTS=(OWRONLY,OCREAT,OTRUNC),
// PATHMODE=SIRWXU
//STDERR DD PATH='/u/usr_id/stderr.f',PATHOPTS=(OWRONLY,OCREAT,OTRUNC),
// PATHMODE=SIRWXU
```

In the CONTROL-M job definition that submits the JCL, use the SET VAR parameter to assign a value to the %%MYSCRPT AutoEdit variable, as follows:

```
SET VAR %%MYSCRIPT=uss script name
```

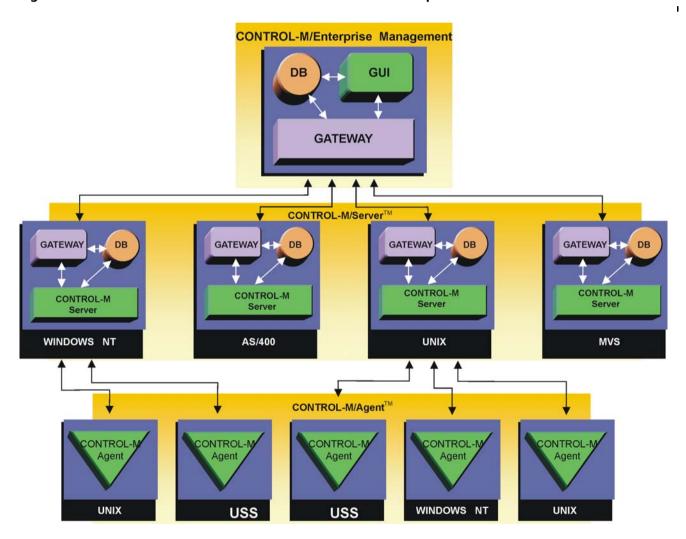
# **Unix Oriented Implementation**

To enable Unix operators to use their expert skills, CONTROL-M provides a Unixoriented MVS implementation for USS jobs.

CONTROL-M incorporates 3-tier architecture that includes CONTROL-M/Enterprise Manager, CONTROL-M/Server, and CONTROL-M/Agent platforms. CONTROL-M treats USS as an additional supported platform, just like any other supported Unix platform.

This architecture is illustrated in Figure 373.

Figure 373 CONTROL-M Architecture for Unix-Oriented MVS Implementation



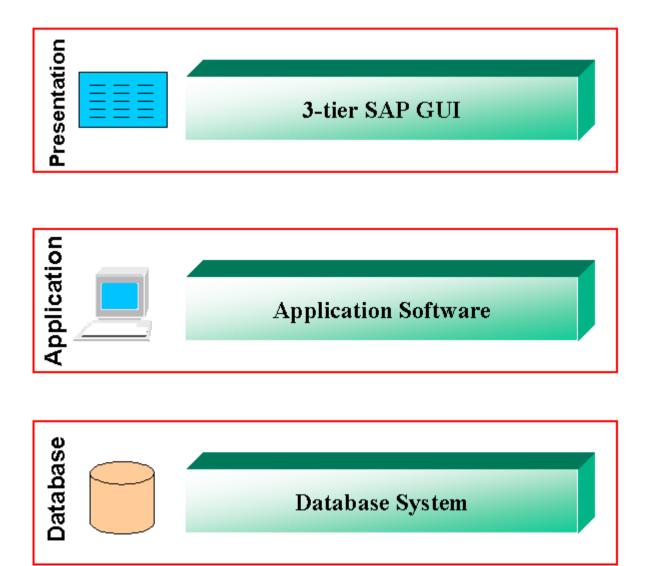
If you have a CONTROL-M/Agent installed on several Unix computers, you can also install it on a USS computer, using the same architecture. You can then implement CONTROL-M in USS through a CONTROL-M/Agent that allows Unix operators to use the tools with which they are familiar.

# **Integrating SAP R/3 running on USS**

In many data centers, the heaviest batch application running on USS is SAP R/3.

The architecture of SAP is shown in Figure 374.

Figure 374 Architecture of SAP R/3



You can integrate this 3-tier architecture with the CONTROL-M MVS platform in the following ways:

- You can use DB/2 as the SAP database, with MVS running the entire SAP database layer. Many users of SAP employ this configuration.
- You can install both of the following in USS:
  - the database layer, using DB/2 running under MVS
  - the application layer

This configuration is popular among organizations that require the stability, scalability, and security of the MVS platform.

## **CONTROL-M Support for SAP in the USS Environment**

CONTROL-M support for the USS environment ensures complete automation and integration of business processes both inside and outside the SAP application environment.

CONTROL-M/Enterprise Management for distributed systems supports SAP R/3 by means of the CONTROL-M Option for SAP R/3. This product is certified by SAP in accordance with the SAP Complementary Software Program.

The combination of CONTROL-M with the CONTROL-M Option for SAP R/3 enables all jobs and tasks to be managed in the same way, regardless of whether the task is

- MVS JCL
- a Unix script
- an SAP task

The SAP R/3 standard Business Application Program Interface (BAPI) enables you to define jobs through either the R/3 or CONTROL-M job definition process.

Once installed on a Windows NT or Unix platform, CONTROL-M Option for R/3 communicates with the R/3 Application layer. The database location is totally transparent to CONTROL-M.

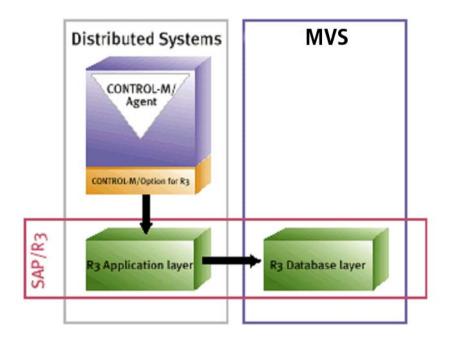
The Application layer can be in either

- SAP R/3
- USS

### **SAP R/3 Application Layer**

If the Application layer is in SAP R/3 in a Unix computer, the communication process between CONTROL-M Option for SAP R/3 and the R/3 Application layer is as shown in Figure 375. In this figure, the database is an MVS (DB/2) database.

Figure 375 Communication with the R/3 Application Layer - DB/2 Database



## **USS Application Layer**

If the Application layer is in USS, use the same Windows NT or Unix CONTROL-M Option for SAP R/3.

CONTROL-M Option for SAP R/3 communicates with the R/3 Application layer in the same way that the product communicates with other platforms. The communication process is shown in Figure 376.

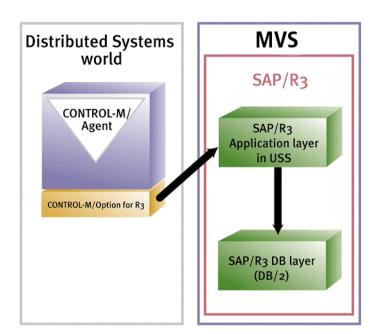


Figure 376 Communication with the R/3 Application - SAP/R3 Database





# Editing Job Scheduling Definitions in the Edit Environment

Job scheduling definition parameters can be edited, moved, copied, deleted, or repeated, by performing IOA Line Editing commands, similar to standard ISPF line commands, from within the Edit environment.

The Edit environment in the Job Scheduling Definition screen is accessed by typing EDIT in the COMMAND field and pressing **Enter**.

Figure 377 The Edit Environment in The Job Scheduling Definition Screen

```
JOB: BACKPO2 LIB CTM.PROD.SCHEDULE
                                                                      TABLE: BACKUP
COMMAND ===>
                                                                     SCROLL===> CRSR
MEMNAME BACKPO2 MEMLIB CTM.PROD.JOBLIB
OWNER M44 TASKTYPE JOB PREVENT-NCT2 Y DFLT N
RPPL APPL-L GROUP BKP-PROD-L
DESC DAILY BACKUP OF SPECIAL FILES FROM APPL-L
__ DESC
__ OVERLIB CTM.OVER.JOBLIB
                                                                  STAT CAL
 __ SCHENV
                                     SYSTEM ID
                                                                  NJE NODE
 __ SET VAR
 ____ CTB STEP AT NAME TYPE
____ DOCMEM BACKPO2 DOCLIB CTM.PROD.DOC
   DAYS
                                                                      DCAL
                                                                            AND/OR
                                                                      WCAL
  __ CONFCAL WORKDAYS SHIFT
                                  RETRO N MAXWAIT OO D-CAT
__ MINIMUM
                    PDS
__ DEFINITION ACTIVE FROM
                                      UNTIL
            START-DAILY-BACKUP ODAT
   ΤN
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
                                                                             16.44.31
```

A 2-character Line Editing command field, marked by underscores, is displayed for each line on the screen.

Editing commands are typed directly onto these underscores.

#### NOTE



Edit commands cannot be applied to the first line of an IN block or an OUT block.

Incorrectly specified Line Editing commands can be corrected by typing over them correctly. Line Editing commands can be deleted by blanking them out or by specifying the RESET command in the COMMAND field.

The Line Editing commands you enter are processed when **Enter** is pressed.

CONTROL-M performs Automatic Syntax Checking to ensure that the job scheduling definition is still syntactically correct after editing. If an edit may invalidate the job scheduling definition, a message is displayed at the top of the screen and the edit is not performed. For recommendations for editing job scheduling definitions, see "Maintaining Valid Job Scheduling Definitions" on page 917.

All operations available in the Job Scheduling Definition screen can be performed while in the Edit environment. For example, parameter values can be changed, and the job scheduling definition can be saved and exited.

To exit the Edit environment, retype EDIT in the COMMAND field and press **Enter**. Line Editing command fields are removed from the display.

Line Editing commands can be performed on the following:

Table 292 Subjects of Line Editing Commands (part 1 of 2)

Subject	Description	
Single Lines	One single line on the screen.	
	Examples:	
	<ul><li>Additional lines of the IN parameter.</li></ul>	
	■ Single-lined DO statement (such as DO COND).	
Logical Lines	All parameter lines for a specific parameter, including its subparameters.	
	Examples:	
	<ul> <li>A DO SHOUT statement, whose subparameters are distributed over more than one line.</li> <li>A single-lined DO statement, such as DO COND.</li> </ul>	

Table 292 Subjects of Line Editing Commands (part 2 of 2)

Subject	Description	
Logical Blocks	Functional group of parameter lines. Job scheduling definitions consist of at least one logical block – an ON block.	
	<b>Example</b> : ON block, which consists of its respective parameter lines and the DO statement lines.	
Multiple Lines	User-specified group of parameter lines.  Example: A series of DO statements.	

# **Line Editing Commands**

The following types of line editing commands exist in the Edit environment.

**Table 293 Line Editing Commands - Delete Commands** 

Command	Description
DS	Delete a single line.
DL	Delete a logical line.
DB	Delete a logical block or sub-block.
DD	Delete lines between two DD specifications.
D	Delete a line. CONTROL-M determines whether to delete a single or logical line based on the line type.

**Table 294 Line Editing Commands - Copy Commands** 

Command	Description	
CS	Copy a single line.	
CL	Copy a logical line.	
СВ	Copy a logical block or sub-block.	
CC Copy lines between two CC specifications.		
C Copy a line. CONTROL-M determines whether to copy a single logical line based on the line type.		
Copy commands are used in conjunction with Location commands. The lines and blocks placed at the position indicated by Location command A or B (described below).		

**Table 295 Line Editing Commands - Move Commands** 

Command	Description	
MS	Move a single line.	
ML	Move a logical line.	
MB	Move a logical block or sub-block.	
MM	Move lines between two MM specifications.	
M	Moves a line. CONTROL-M determines whether to move a single or logical line based on line type.	

Move commands are used in conjunction with Location commands. The lines and blocks are placed at the position indicated by Location command A or B, which are described in Table 298 on page 916.

**Table 296 Line Editing Commands - Repeat Commands** 

Command	Description
RS	Repeat a single line.
RL	Repeat a logical line.
RB	Repeat a logical block or sub-block.
RR	Repeat lines between two RR specifications.
R Repeat a line. CONTROL-M determines whether to repeat a sological line based on line type.	
The repeated lines and blocks are placed immediately after the lines and blocks marked	
the command.	

**Table 297 Line Editing Commands - Insert Command** 

Command	Description	
	Inserts a new logical line or block after the logical line or block marked with an I.	

#### **Table 298 Line Editing Commands - Location Commands**

Command	Description	
Indication of the p	e position where lines or blocks must be placed.	
A (After)	Indicates that lines or blocks must be placed after the line marked with an A.	
B (Before)	Indicates that lines or blocks must be placed before the line marked with a B.	
Location commands A and B are used in conjunction with Copy (C, CS, CL, CC, CB), and Move (M, MS, ML, MM, MB) commands.		

# **Maintaining Valid Job Scheduling Definitions**

Since job scheduling definitions must be syntactically correct at all times, the user must consider the following issues when specifying Line Editing commands:

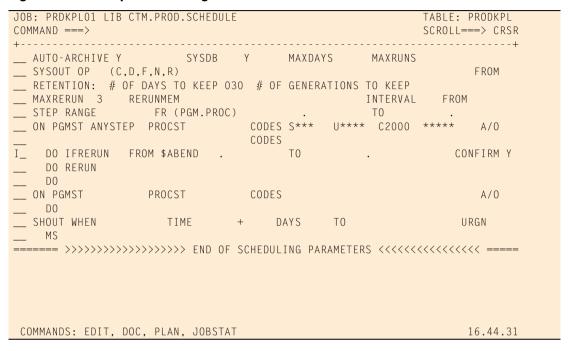
- The result of a Line Editing command is dependent on the line on which the command is specified. For example, command D deletes either a single or a logical line based on the line type.
- Logical lines form a unit and cannot be separated.

When a logical command is specified within a logical line, that is, on a subparameter line or an additional parameter line, the specified operation is performed on the entire logical line.

- Block commands must be specified on the main lines of the block. For example, to delete an ON block, specify command DB (Delete Block) on the ON line.
- Blank parameter lines are added automatically by CONTROL-M, to allow the user to specify additional parameters, and cannot be deleted.
- BMC Software recommends that, wherever possible, you use commands D, C, R, and M for editing, instead of DS, DL, CS, CL, RS, RL, MS, and ML, because these commands automatically retain the logical structure of the job scheduling definition.

**Before:** Insert additional DO statements within a DO block using command I (Insert).

Figure 378 Example - Inserting A DO Statement - Before



#### **After**

Figure 379 Example - Inserting A DO Statement - After

```
JOB: PRDKPL01 LIB CTM.PROD.SCHEDULE
                                                TABLE: PRODKPL
COMMAND ===>
                                                SCROLL===> CRSR
+------
__ AUTO-ARCHIVE Y SYSDB Y MAXDAYS
                                         MAXRUNS
__ SYSOUT OP (C,D,F,N,R)
                                                      FROM
__ RETENTION: # OF DAYS TO KEEP 030 # OF GENERATIONS TO KEEP
___ MAXRERUN 3 RERUNMEM INTERVAL __ STEP RANGE FR (PGM.PROC) . TO
__ ON PGMST ANYSTEP PROCST CODES S*** U**** C2000 ***** A/O
                           CODES
   DO IFRERUN FROM $ABEND .
                               Τ0
                                                   CONFIRM Y
   DO
   DO RERUN
   D0
           PROCST CODES
  ON PGMST
                                                      A/0
   D0
SHOUT WHEN
           TIME + DAYS TO
                                                    URGN
====== >>>>>>>>>>>>> END OF SCHEDULING PARAMETERS <<<<<<<<<<
COMMANDS: EDIT, DOC, PLAN, JOBSTAT
                                                     14.49.42
```

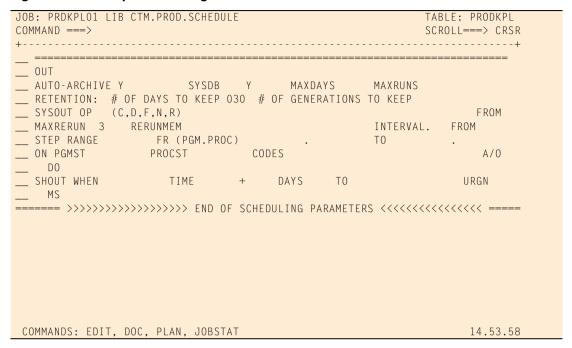
**Before:** Delete an ON PGMST block. Use of the DB (Delete Block) command is the preferred method. The DB command removes all parameters, comments, continuation lines, and separator lines of the specified block. DB must be specified on a main line of the block, that is, ON PGMST. In this example, the ON PGMST block is deleted.

Figure 380 Example - Deleting A Block - Before

	TABLE: PRODKPL SCROLL===> CRSR
OUTAUTO-ARCHIVE Y SYSDB Y MAXDAYS MAXRUNS RETENTION: # OF DAYS TO KEEP 030 # OF GENERATIONS TO KEEP SYSOUT OP (C,D,F,N,R) MAXRERUN 3 RERUNMEM INTERVAL STEP RANGE FR (PGM.PROC) . TO DB ON PGMST ANYSTEP PROCST CODES S*** U**** C2000 ** CODES DO IFRERUN FROM \$ABEND . TO .	
DO ON PGMST PROCST CODES DO	A/0
SHOUT WHEN TIME + DAYS TO MS	URGN
COMMANDS: EDIT, DOC, PLAN, JOBSTAT	14.52.02

#### After: The ON PGMST ANYSTEP block has been deleted.

#### Figure 381 Example - Deleting A Block - After



**Before:** Move multiple DO statements from one sub-block to another. Command MM (Multiple Move) is specified at the beginning and end of the DO statements that are moved. Command A (After) specifies the location after which these lines are placed.

Figure 382 Example - Moving Statements - Before

JOB: PRDKPLO1 LIB CTM.PROD.SCHEDULE	TABLE: PRODKPL
COMMAND ===>	SCROLL===> CRSR
	+
OUT	
AUTO-ARCHIVE Y SYSDB Y	
RETENTION: # OF DAYS TO KEEP 030	
SYSOUT OP (C,D,F,N,R)	FROM
MAXRERUN 3 RERUNMEM STEP RANGE FR (PGM.PROC)	INTERVAL. FROM
ON DOMST ANYSTED DOGCST	TO . CODES S*** U**** C2000 ***** A/O
	CODES
DO IFRERUN FROM \$ABEND .	TO . CONFIRM Y
DO RERUN	
MM DO COND STEP5_DONE	ODAT +
DO SHOUT TO TSO-M22	URGENCY R
MM = STEP STEP05 PROCESSED	
D0	
ON PGMST STEP05 PROCST	
A_ DO IFRERUN FROM \$ABEND .	TO . CONFIRM N
DO	CODEC
ON PGMST PROCST	CODES A/O
SHOUT WHEN TIME +	DAYS TO URGN
COMMANDS: EDIT, DOC, PLAN, JOBSTAT	15.03.25

**After:** The two specified DO statements have been moved to the specified location.

Figure 383 Example - Moving Statements - After

JOB: PRDKPL01 LIB CTM.PROD.SCHEDULE COMMAND ===> +	TABLE: PRODKPL SCROLL===> CRSR
OUT AUTO-ARCHIVE Y SYSDB Y MAXDAYS MAXRUNS	
RETENTION: # OF DAYS TO KEEP 030 # OF GENERATIONS TO KEEP	
SYSOUT OP (C,D,F,N,R)	FROM
MAXRERUN 3 RERUNMEM INTERVAL.	FROM
STEP RANGE FR (PGM.PROC) . TO ON PGMST ANYSTEP PROCST CODES S*** U**** C2000	·
UN PGMST ANTSTEP PROCST CODES SAAA UAAAA CZUUU	A/U
DO IFRERUN FROM \$ABEND . TO .	CONFIRM Y
DO RERUN	
DO ON PGMST STEP05 PROCST CODES S***	A/0
ON PGMST STEPUS PROCST CODES S^^^ _ DO IFRERUN FROM \$ABEND . TO .	CONFIRM N
DO COND STEP5_DONE ODAT +	00111 21111 11
DO SHOUT TO TSO-M22 URGENCY R	
<pre>= STEP STEP05 PROCESSED D0</pre>	
ON PGMST PROCST CODES	A/0
D0	,,, 0
SHOUT WHEN TIME + DAYS TO	URGN
COMMANDS: EDIT, DOC, PLAN, JOBSTAT	5.06.09

**Before:** Copy ON PGMST and some of its DO statements to another ON PGMST block. Command CC (Multiple Copy) is specified at the beginning and the end of the parameters that is copied. Command B (Before) specifies the location before which these lines are placed.

Figure 384 Example - Copying Statements - Before

JOB: PRDKPLO1 LIB CTM.PROD.SCHEDULE	TABLE: PRODKPL
COMMAND ===>	SCROLL===> CRSR
AUTO-ARCHIVE Y SYSDB Y MAXDAYS MAXRUNS RETENTION: # OF DAYS TO KEEP 030 # OF GENERATIONS TO KEEP SYSOUT OP (C,D,F,N,R) MAXRERUN 3 RERUNMEM INTERVAL	FROM
MAXRERON 3 RERONMEN INTERVAL STEP RANGE FR (PGM.PROC) . TO	
ON PGMST ANYSTEP PROCST CODES S*** U**** C2000 CODES	**** A/0
DO IFRERUN FROM \$ABEND . TO DO RERUN DO	CONFIRM Y
CC ON PGMST STEP05 PROCST CODES S***  DO IFRERUN FROM \$ABEND . TO .  CC DO COND STEP5_DONE ODAT +  DO SHOUT TO TSO-M22 URGENCY R  = STEP STEP05 PROCESSED  DO	A/O CONFIRM N
B ON PGMST PROCST CODES DO	A/0
SHOUT WHEN TIME + DAYS TO MS	URGN
COMMANDS: EDIT, DOC, PLAN, JOBSTAT	14.32.29

After: The specified ON PGMST and DO statements have been copied.

Figure 385 Example - Copying Statements - After

JOB: PRDKPL01 LIB CTM.PROD.SCHEDULE COMMAND ===>		TABLE: PRODKPL SCROLL===> CRSR
MAXRERUN 3 RERUNMEM RETENTION: # OF DAYS TO KEEP 030	INTERVAL. # OF GENERATIONS TO KEEP	FROM
STEP RANGE FR (PGM.PROC) ON PGMST ANYSTEP PROCST	CODES S*** U**** C2000 CODES	***** A/0
DO IFRERUN FROM \$ABEND DO RERUN DO	ТО .	CONFIRM Y
ON PGMST STEP05 PROCST		A/0
DO IFRERUN FROM \$ABEND DO COND STEP5_DONE DO SHOUT TO TSO-M22 = STEP STEP05 PROCESSED DO	ODAT +	CONFIRM N
ON PGMST STEP05 PROCST		A/0
DO IFRERUN FROM \$ABEND DO COND STEP5_DONE DO		CONFIRM N
ON PGMST PROCST	CODES	A/0
COMMANDS: EDIT, DOC, PLAN, JOBSTAT		15.19.53

**Before:** Insert a continuation line between existing continuation lines. It is recommended that command RS (Repeat Single) or R (Repeat) be used to repeat the previous line.

Figure 386 Example - Inserting A Line - Before

JOB: PRDKPL01 LIB CTM.PROD.SCHEDULE TABLE: PRODKPL
COMMAND ===> CRSR
TIME 70ME
TIME ZONE:
===================================
AUTO-ARCHIVE Y SYSDB Y MAXDAYS MAXRUNS
RETENTION: # OF DAYS TO KEEP 030 # OF GENERATIONS TO KEEP
SYSOUT OP (C,D,F,N,R) FROM
MAXRERUN 3 RERUNMEM INTERVAL. FROM
STEP RANGE FR (PGM.PROC) . TO .
ON PGMST ANYSTEP PROCST CODES S*** U**** C2000 C3000 A/O A
RS
ON PGMST STEP04 PROCST CODES ***** A/O
DO IFRERUN FROM \$ABEND . TO . CONFIRM Y
DO RERUN
DO
ON PGMST STEP05 PROCST CODES S***
DO IFRERUN FROM \$ABEND . TO . CONFIRM N
DO COND STEP5_DONE ODAT +
DO SHOUT TO TSO-M22 URGENCY R = STEP STEP05 PROCESSED
COMMANDS: EDIT, DOC, PLAN, JOBSTAT 15.22.46

**After:** The continuation line has been repeated. The repeated line can be modified as necessary.

Figure 387 Example - Inserting A Line - After

JOB: PRDKPL01 LIB CTM.PROD.SCHEDULE COMMAND ===>					SCROLL=	PRODKPL > CRSR
TIME ZONE:						+
===================================						
AUTO-ARCHIVE Y SYSDB						
RETENTION: # OF DAYS TO KEEP 030	# 0F	GENERA	TIONS	ro keep		
SYSOUT OP (C,D,F,N,R)						FROM
MAXRERUN 3 RERUNMEM					L. FRC	)M
STEP RANGE FR (PGM.PROC)			I I de la desta de la de	T0		
ON PGMST ANYSTEP PROCST						A/0 A
		C4000	C5000			
_		C4000	C5000	C6000	C7000	
 ON PGMST STEP04 PROCST		C1200				A/0
ON PGMST STEP04 PROCST DO IFRERUN FROM \$ABEND .					400	IFIRM Y
DO IFRERON FROM \$ABEND . DO RERUN		10			CUN	ILIKM I
DO RERON DO						
ON PGMST STEPO5 PROCST	CODES	C***				A/0
DO IFRERUN FROM \$ABEND .					COV	IFIRM N
DO COND STEP5 DONE					CON	11 1 1 1 1 1 1
DO SHOUT TO TSO-M22						
COMMANDS: EDIT, DOC, PLAN, JOBSTAT	0	NGENOI				15.23.46





# **Editing CMEM Rule Definitions in the Edit Environment**

CMEM rule definition parameters can be edited (moved, copied, deleted, repeated) by performing IOA Line Editing commands, similar to standard ISPF line commands, from within the IOA Edit environment.

The Edit environment in a Rule Definition screen is accessed by typing EDIT in the COMMAND field and pressing **Enter**.

Figure 388 The Edit Environment in The Rule Definition Screen

A 2-character Line Editing command field, marked by underscores, is displayed for each line on the Rule Definition screen.

Editing commands are typed directly onto these underscores.

Incorrectly specified Line Editing commands can be corrected by typing over them correctly. Line Editing commands can be deleted by blanking them out or by specifying the RESET command in the COMMAND field.

The Line Editing commands you enter are processed when **Enter** is pressed.

The CMEM facility performs Automatic Syntax Checking to ensure that the rule definition is still syntactically correct after editing. If an edit may invalidate the rule definition, a message is displayed at the top of the screen and the edit is not performed. For guidelines and recommendations for editing rule definitions, see "Maintaining Valid Rule Definitions" on page 930.

All operations available in the Rule Definition screen can be performed while in the Edit environment. For example, parameter values can be changed, and the Rule Definition screen can be saved and exited.

To exit the Edit environment, re-type EDIT in the COMMAND field and press **Enter**. Line Editing command fields are removed from the display.

Line Editing commands can be performed on any single ON or DO statement or on a block of ON or DO statements.

All lines of a single statement, for example, the two lines of a DO FORCEJOB statement, constitute a logical line.

# **Line Editing Commands**

The following types of line editing commands exist in the Edit environment.

**Table 299 Line Editing Commands - Delete Commands** 

Command	Description
DS	Delete a single line.
DL	Delete a logical line.
DB	Delete a logical block or sub-block.
DD	Delete lines between two DD specifications.
D	Delete a line. CONTROL-M determines whether to delete a single or logical line based on the line type.

Table 300 Line Editing Commands - Copy Commands (part 1 of 2)

Command	Description
CS	Copy a single line.
CL	Copy a logical line.

Table 300 Line Editing Commands - Copy Commands (part 2 of 2)

Command	Description
СВ	Copy a logical block or sub-block.
CC	Copy lines between two CC specifications.
С	Copy a line. CONTROL-M determines whether to copy a single or logical line based on the line type.
Copy commands are used in conjunction with Location commands. The lines and blocks are placed at the position indicated by Location command A or B (described below).	

**Table 301 Line Editing Commands - Move Commands** 

Command	Description
MS	Move a single line.
ML	Move a logical line.
MB	Move a logical block or sub-block.
MM	Move lines between two MM specifications.
M	Moves a line. CONTROL-M determines whether to move a single or logical line based on line type.
Move comman	ds are used in conjunction with Location commands. The lines and blocks are

Move commands are used in conjunction with Location commands. The lines and blocks are placed at the position indicated by Location command A or B, described in Table 304 on page 930.

**Table 302 Line Editing Commands - Repeat Commands** 

Command	Description
RS	Repeat a single line.
RL	Repeat a logical line.
RB	Repeat a logical block or sub-block.
RR	Repeat lines between two RR specifications.
R	Repeat a line. CONTROL-M determines whether to repeat a single or logical line based on line type.
The repeated li the command.	nes and blocks are placed immediately after the lines and blocks marked with

**Table 303 Line Editing Commands - Insert Command** 

Command	Description
I	Inserts a new logical line or block after the logical line or block marked with an I.

**Table 304 Line Editing Commands - Location Commands** 

Command	Description
Indication of the position where lines or blocks must be placed.	
A (After)	Indicates that lines or blocks must be placed after the line marked with an A.
B (Before)	Indicates that lines or blocks must be placed before the line marked with a B.
Location commands A and B are used in conjunction with Copy (C, CS, CL, CC), and Move (M, MS, ML, MM) commands.	

# **Maintaining Valid Rule Definitions**

Since rule definitions must be syntactically correct at all times, you must consider the following issues when specifying Line Editing commands:

- The result of a Line Editing command is dependent on the line on which the command is specified. For example, command D deletes either a single or a logical line based on the line type.
- Logical lines function as a unit and cannot be separated.

When a logical command is specified within a logical line, that is, on a subparameter line, or a continuation line, the specified operation is performed on the entire logical line.

- Block commands must be specified on the main lines of the block. For example, to delete an ON block, specify command DB (Delete Block) on the ON line.
- Blank parameter lines added automatically by CMEM, to allow the user to specify additional parameters, cannot be deleted.
- BMC Software recommends that, wherever possible, you use commands D, C, R, and M for editing, instead of DS, DL, CS, CL, RS, RL, MS, and ML, because these commands automatically retain the logical structure of the rule definition.

**Before:** Repeat a DO block in the Rule Definition screen.

Figure 389 Example - Repeating A DO Block - Before

```
RL: JOBNAM1
            LIB CTM.PROD.RULES
                                                       TABLE: CMEMRULE
COMMAND ===>
                                                       SCROLL==> CRSR
                                                       . - - - - - - - - - - - - +
___ ON JOBARRIV = JOBNAM1 JTYPE SMFID SYSTEM And/Or/Not 
___ OWNER CTMCTLM GROUP MODE PROD RUNTSEC NONE
___ DESCRIPTION CONVERSION: ON JOB JOBNAM1 ARRIVAL FORCEJOB
__ DESCRIPTION
R_ DO FORCEJOB = TABLE TABLE1 JOB
                                                 DATE ODAT
             LIBRARY CTM.PROD.SCHEDULE
__ DO
 _ -----
----- >>>>>>>> END OF RULE DEFINITION PARAMETERS <<<<<<<<<
FILL IN RULE DEFINITION. CMDS: CAPS, EDIT, SHPF,
                                                             20.10.46
```

After: The DO block has been repeated.

Figure 390 Example - Repeating A DO Block - After

```
LIB CTM.PROD.RULES
RI: JOBNAM1
                                                       TABLE: CMEMRULE
COMMAND ===>
                                                       SCROLL===> CRSR
 ON JOBARRIV = JOBNAM1 JTYPE SMFID SYSTEM
OWNER CIMCILM GROUP
                                                     And/Or/Not
                                        MODE PROD RUNTSEC NONE
___ OWNER CTMCTLM GROUP
___ DESCRIPTION CONVERSION: ON JOB JOBNAM1 ARRIVAL FORCEJOB
__ DESCRIPTION
__ DO FORCEJOB = TABLE TABLE1 JOB
                                                       DATE ODAT
              LIBRARY CTM.PROD.SCHEDULE
  DO FORCEJOB = TABLE TABLE1 JOB
                                                      DATE ODAT
      LIBRARY CTM.PROD.SCHEDULE
----- >>>>>>>> END OF RULE DEFINITION PARAMETERS <<<<<<<<<<
 FILL IN RULE DEFINITION. CMDS: CAPS, EDIT, SHPF, 20.32.47
```





# MVS Job Restart Without CONTROL-M/Restart

For sites in which CONTROL-M/Restart is not installed, CONTROL-M provides a mechanism for automatic implementation of MVS restarts in certain situations. The mechanism, however, requires definition before the original submission of the job. Therefore, it is only useful for jobs in which automatic restart is always desirable (when necessary).

#### NOTE -



MVS restart is not recommended and must be used with caution. MVS restart does not perform automatic File catalog adjustment, GDG (Generation Dataset Group) adjustment, condition code recapture, abend code recapture, or data set scratching. Unless these functions are manually handled without error, the results of an MVS restart may be unpredictable or damaging.

The mechanism for automatic implementation of MVS restart is the definition of a special OUT condition in the job scheduling definition. The value of the condition is:

@非- ODAT

#### where:

- @# =condition
- $\blacksquare$  *ODAT* = date
- -=option

#### NOTE -



**§Restart§** Do not define this type of restart (that is, this OUT condition) if a CONTROL-M/Restart restart is used for the job, or the results may be unpredictable. **§Restart§** 

This restart is implemented in the following situations if the CONTROL-M monitor ended the job NOT OK (that is, a DO OK did not impact the final status):

- The job abended.
- The job failed due to JCL error.
- One of the steps ended with a condition code of C2000 (abend of a PL/1 program).

When the special OUT condition is defined in the job scheduling definition and the job ends as described above, the CONTROL-M monitor automatically appends the name of the failing step to the OUT condition of the job order. The OUT condition in the job order, that is, as seen in the Zoom screen, therefore appears as follows:

#### @非-procstep.pgmstep

Before a job is submitted, the CONTROL-M monitor checks the job order for an OUT condition beginning @#-. When the monitor detects condition @#-procstep.pgmstep, it automatically inserts an MVS step in the JCL of the job, so that the job begins from the indicated *procstep.pgmstep*.

For the job to be restarted from *procstep.pgmstep*, the job must be rerun. This can be the result of a rerun, manual or automatic, or the result of a cyclic job run.

The @#-procstep.pgmstep value appearing in the Zoom screen can be deleted, in which case restart is not performed, or changed to a different procstep.pgmstep, so that restart begins from a different step.

Even if a special OUT condition (name or prefix @#-) is not defined in the job scheduling definition, an MVS restart can be implemented by specifying OUT condition @#- procstep.pgmstep (for the desired restart step) in the Zoom screen.

#### – NOTE –



It is also possible to specify OUT condition @#- procstep.pgmstep in the job scheduling definition, but this is not recommended. If @#- procstep.pgmstep is specified in the job scheduling definition, the job always begins at the specified step, never at the first step, even on the initial run.

When using MVS job restart, every step in the job must have a unique *procstep.pgmstep* name. CONTROL-M does not check for duplicate stepnames.

The following is an example of a job set for Automatic Restart, using CONTROL-M only, in case of abend.

#### NOTE —



**§Restart§** Do not use this type of restart when CONTROL-M/Restart restart is used for the job, or results may be unpredictable.

Figure 391 Example - Automatic Restart - CONTROL-M Only

```
JOB: EBDUPDT2 LIB CTM.PROD.SCHEDULE
                                                                     TABLE: EBDPROD
COMMAND ===>
                                                                     SCROLL===> CRSR
  MEMNAME EBDUPDT2 MEMLIB GENERAL

OWNER SYS1 TASKTYPE JOB PREVENT-NCT2 DFLT N

APPL EBD GROUP FRD-PRODUCTION
  APPL EBD GROUP
DESC EBD PRODUCTION UPDATE OF DEPOSITS
                                         GROUP EBD-PRODUCTION
  OVERLIB
                                                                  STAT CAL
  SCHENV
                                    SYSTEM ID
                                                                  NJE NODE
  SET VAR
  SET VAR
CTB STEP AT NAME TYPE
DOCMEM EBDUPDT2 DOCLIB CTM.PROD.DOC
  DAYS
                                                                     DCAL
                                                                         AND/OR
  WDAYS 2,3,4,5,6
                                                                     WCAL
  MONTHS 1- Y 2- Y 3- Y 4- Y 5- Y 6- Y 7- Y 8- Y 9- Y 10- Y 11- Y 12- Y
  DATES
  CONFCAL
                   SHIFT
                               RETRO Y MAXWAIT 08 D-CAT
  CONFCAL SHIF MINIMUM PDS
  DEFINITION ACTIVE FROM
                                 UNTIL
  IN DEPOSITS
                                 PREV
  CONTROL
  RESOURCE
  PIPE
  FROM TIME + DAYS UNTIL TIME + DAYS
DUE OUT TIME + DAYS PRIORITY SAC CONFIRM
  TIME ZONE:
                        0DAT + @#-
                                                                   ODAT -
            DEPOSITS
 COMMANDS: EDIT, DOC, PLAN, JOBSTAT
                                                                           11.17.00
```



# Index

# **Symbols**

L Cl	DOIFRERUN 448
! Character	SFIRST.\$CLEANUP
CTMQSB Command 813	DO IFREFUN Statement 448
Hex Value 82	% (Simulation) Option
Character	Active Environment Screen 183
Prerequisite Condition 562	% Symbol
# Character	Job Graph 205
Maybe Job Prefix 819	SHOUT Parameter 629
User-Defined Variable 754	%%
# OF DAYS TO KEEP	Concatenation Symbol 769
RETENTION Parameter 603, 605	%% Prefix
# OF GENERATIONS TO KEEP	User Defined Variable 754
RETENTION Parameter 603, 605	
#JNFRT Parameter, CTMPARM 217	%% Symbol
#WSC Field, Global View Screen 201	AutoEdit Term 46, 745
\$ Character	Calendar Date 750
Hex Value 82	Concatenation 748, 769
Job Graph 205	%%\$ARGS
RESOURCE Parameter 599	CONTROL-O System Variable 708
User-Defined Variable 754	%%\$CALCDTE Function
\$\$\$\$ Date Reference	Comparison with %%CALCDATE 78
IN Parameter 501	Description 781
OUT Parameter 563	Example 801
\$\$ACTDOC Member	%%\$CENT
Customization 81	First Two Digits of the Year 751
Customizing Active Environment Screen 169	%%\$COMMSYS Reserved Variable
\$ABEND	DO SHOUT Statement 711
DO IFRERUN Statement 448, 450	%%\$DATEFMT
Restart Confirmation Window 228	Date Format 749
\$CLEANUP Value	%%\$Dn
Rerun/Restart Window 228	CMEM AutoEdit Variable 682, 701
\$D MEMBER Command	%%\$DSN
Machine ID Under JES2 472, 633, 713	CMEM AutoEdit Variable 682
\$DEFAULT	%%\$DSNDISP
Step Adjustment 227	CMEM AutoEdit Variable 683
\$DEFAULT Member	%%\$GREG Function
Restart Window 226	Julian to Gregorian Conversion 783
\$EXERR	%%\$GRID
DO IFRERUN Statement 448, 450	Resolution 753
Restart Confirmation Window 228	Special AutoEdit System Variable 753
SFIRST	%%\$JNAME
DO IFRERUN Statement 448, 450	CMEM AutoEdit Variable 683, 701
Restart Confirmation Window 228	%%\$JULIAN Function
\$FIRST Value	Gregorian to Julian Conversion 783
Rerun/Restart Window 228	%%\$L
SFIRST.SABEND	LPAR name 749
Y	%%\$LEAP Function

T 77 A 1 1 704	G . D . ME1 ME0
Leap Year Analysis 784	System Date 751, 752
%%\$LENGTH Function	%%DAY
Length Extraction 789	System Day 751
%%\$MEMNAME	%%ELSE Control Statement
System Variable 749	Example 808
%%\$OCENT	JCL Setup 772
System Variable 751	%%ENDIF Control Statement
%%\$ORDERID	Example 808
AutoEdit Simulation 329	JCL Setup 772
%%\$QNAME	%%GLOBAL
Monitor Identifier 749	AutoEdit Statement 331, 746
%%SRCENT	JCL Setup 766, 771
System Variable 751	Member Format 759
%%\$RJULDAY	%%GLOBAL Control Statement
Julian Working Day 752	Local Variable 758
%%\$SABEND	%%GLOBAL Members
CMEM AutoEdit Variable 683	Cache 758
%%\$SCHDLIB	%%GOTO Control Statement
System Variable 749	Example 809
%%\$SCHDTAB	JCL Setup 772
System Variable 749	%%GROUP
%%\$SIGN	Job Group 749
Quantitative Resource 750	%%IF Control Statement
%%\$STEPCC	Example 808
CMEM AutoEdit Variable 683	JCL Setup 772
%%\$SYSNAME	Nesting 774
System Variable 750	%%INCLIB Control Statement
%%\$TAG	Example 808
AutoEdit Simulation 329	JCL Setup 775
Schedule Tag Name 750	%%INCMEM Control Statement
%%\$UABEND	Example 808
CMEM AutoEdit Variable 683	JCL Setup 775
%%\$WCALC Function	%%JOBCC
Working Date Shift 784	Final Job Status 753
%%\$WEEK# Function	Force OK 754
Week of Year 785	%%JOBID
%%\$WEEKDAY Function	JES Job Number 754
Day of Week Analysis 786	%%JOBNAM Variable
%%SYEARWK# Function	SHOUT Statement 361, 369
Week of Year 787	%%JOBNAME
%%A.%%B 767	AutoEdit Symbol 485
%%APPL	%%JOBNAME Variable
Application 748	SHOUT Statement 361
%%BLANK	Submitted Job Name 749
Blank 748	%%JULDAY
Compared with %%BLANKn 749 %%BLANKn	Julian Day 751, 752 %%LABEL Control Statement
Compared with %%BLANK 749	Example 809
n Blanks 749	JCL Setup 772
SET VAR Parameter 621	%%LIBSYM Control Statement 826
%%CALCDATE	AutoEdit Statement 746
Comparison with %%\$CALCDTE 788	JCL Setup 766, 776
%%CALCDATE Function	Local Variable 758
AutoEdit Function 747	PROMPT Library 828
Description 788	%%MAXRC
%%DATE	Force OK 754
Example 797	Highest Return Code 753

%%MEMNAME	%%RDAY
AutoEdit Symbol 485	Working Day 751
%%MEMSYM	%%RESOLVE
Member Format 759	ALLCACHE 759
%%MEMSYM Control Statement 826	%%RESOLVE Control Statement
AutoEdit Statement 746	Example 807
JCL Setup 766, 776	JCL Setup 766, 778
Local Variable 758	%%RESOLVE NO Control Statemen
Table Name Prefix 828	AutoEdit Logic 778
%%MINUS Operator	JCL Setup 780
AutoEdit 781	%%RESOLVE YES Control Statemen
%%MM.%%DD.%%YY	Example 807
Example 801	%%RJULDAY
%%MONTH	Julian Working Day 751
Month of the Job 751	%%RMONTH
System Month 751	Working Month 751
%%O	%%RN
	Run Number 749
ODATE 750	
%%ODATE	%%RWDAY
Date of the Job 751, 752	Working Day of the Week 751
Example 797	%%RWEEK
%%ODAY	Working Week 751
Day of the Job 751	%%RYEAR
Example 798	Working Year 751, 752
%%ODAY.%%A.%%OYEAR	%%SET %%variable
Example 800	AutoEdit Control Statement 746
%%OJULDAY	%%SET Control Statement
Julian Day of the Job 751, 752	Global Variable 760
%%OMONTH	JCL Setup 765, 766, 779
Example 798	Local Variable 757
%%ORDERID	OCCUR NO 837
Job Order ID 749	User-Defined Variable 755
%%OWDAY	Variable Members 766
Day of the Week of the Job 751	%%SIGN
Example 803	System Variable 599
%%OWDAY.%%TIME	%%STEP
Example 806	Latest Program Step 753
%%OWEEK	%%SUBSTR Function
Week of the Job 751	Example 801
%%OWNER	String Extraction 788
Job Owner 749	%%TIME
%%OYEAR	Example 806, 808, 809
Example 798	Time of Day 749
Year of the Job 751, 752	%WDAY
%%PLANID	Day of the Week 751
Inter-plan Dependency 835	%%WEEK
%%PLUS Operator	Week of the Year 751
AutoEdit 747, 781	%YEAR
%%R	
	System Year 751, 752
Installation Working Date 750	'Character
%%RANGE Control Statement	FIND Command 97
Example 806	( Character
JCL Setup 777	Hex Value 82
Minimum Length 777	() Characters
%%RDATE	DO COND 437
Example 798	IN Parameter 505
Working Date 751, 752	Prerequisite Condition 562

* Character	+EVERY Step 549
CONFCAL Calendar 421, 670	+ (Group Next)
D-CAT Parameter 416	SAC Parameter 610
DO SYSOUT Statement 480	+ Sign
JCL 767	Change Resource Window 289
Job Graph 205	DO COND Statement 438
Masking 83	Job Dependency Network 242
MEMNAME Value 705	Job Graph 205
ON PGMST Statement 642	OUT Parameter, OPT Subparameter 563
ON Statement 547	SHOUT Parameter 629
ON Statement Codes 551	+EVERY
PRIORITY Parameter 588	ON Statement 543, 544
Quick Schedule Definition 367	PGMST Parameter 543
SHOUT Parameter 628	PROCST Parameter 544
SYSOUT Parameter 644, 647	+EVERY Step Value
* in DCAL Parameter	ON Statement 548
CONFCAL Calendar 421	+nnn Variable
* in WCAL Parameter	Schedule Date 437, 501, 563
CONFCAL Calendar 670	- (Group Previous)
* Symbol	SAC Parameter 610
SCHEDULE TAG Parameter 501	- Sign
*\$EJ Code	Change Resource Window 289
ON Statement Codes 551	DO COND Statement 438
**** Date Reference	Job Dependency Network 242
IN Parameter 501	OUT Parameter, OPT Subparameter 563
OUT Parameter 563	Quick Schedule Definition 367
Schedule Date 437	SHOUT Parameter 629
***** Code	-nnn Variable
+EVERY Step 549	Schedule Date 437, 501, 563
ON Statement Codes 551	. Character
*.taskid	AutoEdit Variable 767
MEMLIB Parameter 524	Character
*FLUSH Code	AutoEdit Variable 768
ON Statement Codes 552	/* Symbol
*in-condition	DO Statement Command 262
Quick Schedule Definition 367	//*CONTROL-M
*NCT2 Code	Quick Submit Command 814
ON Statement Codes 551	//OUTPUT Statement
*P Field	SYSDATA Output Class 68
Conditions/Resources 283, 290	< Character
*rangename	ON Statement CODES 555
ON Statement 543	SHOUT Parameter 629
PGMST Parameter 543	=6 Command
*REC0 Code	PF06/PF18 93, 109
ON Statement Codes 551	=X Command
*SNRUN Code	Fast Exit 91
ON Statement Codes 553	Online Facility Exit 91
*T Command	> Character
JES Name 470, 630	DO Statement 435
JES3 524	ON Statement 546
*TERM Code	ON Statement CODES 555
ON Statement Codes 551	SHOUT Parameter 629
*UKNW Code	TIME Parameter 659
ON Statement Codes 551	? Character
*UKNW Status	
ON Statement 548	Masking 83
ON Statement 548 *xxxx Code	? Option Active Environment Screen 180, 205
xxxx code	Active Environment Screen 180, 205

? Symbol	ABENDED Status
Confirm Rerun Window 226	Show Screen Filter 195
Restart Window 233	ABORT
@ Character	Screen Command 95
Hex Value 82	ACF2 577
Maybe Job Prefix 819	Action Keyword
OS/390 Restarts 819, 933	DO Statement 434
User-Defined Variable 754	ACTIVATE Option
@ Symbol	Active Environment Screen 181
Maybe Jobs 819	Active Environment Screen
@#-procstep.pgmstep	Commands 174
OS/390 Restart 934	Display Type 169
OUT Condition 934	Fields 171, 173
@#_	Filtering 189
OS/390 Restart 933	Format 170
Character	Functions 166
DO COND Statement 437	Job Deleting 180
Hex Value 82	Job Statuses 185
Prerequisite Condition 562	Options 180
¬ Character	RBA 175
Prerequisite Condition 562	Active Environment screen 176, 216, 462
Trorequisite condition out	ACTIVE FROM Parameter
	Scheduling Logic 382
Numerics	ACTIVE IN ERROR Status
Mullielics	Active Environment Screen 189
1 Command	Active Jobs File
IOA Primary Option Menu 87, 88	Daily Subsystem 379
35-Day Default	Description 45
Periodic Calendar 318	Display 59
4 Option	DO FORCEJOB Statement 704
Primary Option Menu 86	Dynamic Insert Facility 69
5 Option	MAXWAIT Parameter 498, 519
Primary Option Menu 86	New Day Processing 46
6 Option	PRIORITY Parameter 588
Primary Option Menu 86	
8 Option	Restoration 50
Primary Option Menu 86	SYSDATA Deletion 399
Timary Option Wend ov	TASKTYPE Parameter 656 Active Missions File
A	D-CAT Parameter 416
^	ACTIVE Status
A Option	Active Environment Screen 185, 189
Active Environment Screen 181	Group Entity 566
Manual Conditions 294, 818	ACTIVE UNTIL Parameter
Parameter Prompting 344	Scheduling Logic 382
A/O Parameter	ADD Command
ON Statement 544	Conditions and/or Resources 285
ABEND	ADD COND Command
FLUSH 553	Conditions and/or Resources 286
SNRUN 554	ADD COND Parameter
Abend Capturing Option	Simulation 844
DUMP Command 179	Add Condition Option
Abend Code	Why Screen 208
ON Statement 550	ADD CONTROL Command
S*37 624	Conditions and/or Resources 286
	ADD Option
Abend Code Recapture	Manual Conditions 294, 818
Rerun Confirmation 226	Parameter Prompting 344, 355

ADD RESOURCE Command	CMEM Rule Definition 259
Conditions and/or Resources 286	ANYSTEP
ADDED Field	FORCE OK 247
Manual Conditions 292	ON Statement 543
Adding	PGMST Parameter 543
Conditions and/or Resources 285	ANYSTEP Value
Manual Condition 292	PGMST Parameter 548
Adding Variables	APPL FORM Parameter
Variable Database Facility 277	Description 395
Addition Operator	APPL Parameter
%%PLUS Operator 781	Description 393
ADDMNCND Utility	Example 393
Maybe Jobs 820	APPL TYPE Parameter
ADJUST CONDITIONS	Description 396
Parameter 389	APPL VER Parameter
ADJUST CONDITIONS Parameter	Description 397
Description 389	Application Name
Group Entity 111, 140, 377, 820	APPL Parameter 393
AECACHL Parameter	APPL TYPE Parameter 396
CTMPARM 758	APPLICATION NAME Parameter
AELIBNM Parameter	AutoEdit Simulation 330
CTMEXEC CLIST 834	Application Program Interface, CONTROL-M 862
AJF	Application Version
Functions Performed by CTMAPI 865	APPL VER Parameter 397
AJF Action BAPIAACT Field Values 896	Arriving
AJF Action Input Parameters 896	SYSDATA 398
AJF Action Return Codes	
CTMAPI 897	Sysout 484 ARG Parameter
AJF Actions	DO CTBRULE Statement 442
	ARGUMENTS Parameter
CTMAPI Calling 870 ALCMAJF Member 891	CTB STEP Parameter 413
Use with CTMAPI 871	ARROW Command
ALL Argument	Change Color 163 ASK FOR EACH ONE Field
REFRESH Command 176, 245	CMEM Rule Order Window 268
All Info Display Type Active Environment Screen 172	Conditions/Resources Confirmation 288
Active Environment Screen 172 All Runs	Manual Conditions Confirmation 295
ON Statement 546	
STEP RANGE Parameter 643	Scheduling Confirmation 154 Why Screen Confirmation 210
ALLCACHE Value	
%%RESOLVE Control Statement 779	Assignment of Variable
ALLRUNS Parameter	Definition 763 AT Parameter
CTRPARM Member 546, 643	CTB STEP Parameter 413
ALLRUNS=YES	ATTN Key
FLUSH Code 553	AutoRefresh Mode 100
SNRUN Code 554	AUTO Command
AND/OR Parameter	AutoRefresh Mode 100, 176
DAYS/WDAYS Parameter 381, 421, 670	AUTO-ARCHIVE parameter 400
And/Or/Not Logic	AUTO-ARCHIVE parameter and 400
CMEM ON Statement 722	AutoEdit Expression
And/Or/Not Parameter	DO RULE Statement 707
CMEM ON Statement 721	AutoEdit Facility
ON DSNEVENT Statement 726	Boolean Logic 772
ON JOBARRIV Statement 729	Control Statement 770, 835
ON JOBEND Statement 731	JCL Modification 745
ON STEP Statement 734	JCL Setup 46
And/Or/Not Subparameter	Job Scheduling 747, 795

Setting Variable Values 780	Precedence 769
Syntax Checking 328, 793	Resolution 767
AutoEdit Function	SET VAR Parameter 622
%%\$CALCDTE 781	Setting a Value 465, 621, 769
%%\$GREG 783	System Variable 748
%%\$JULIAN 783	Value Assignment 769
%%\$LEAP 784	AutoEdit variable
%%\$LENGTH 789	in condition name 294, 437, 500, 562, 796
%%\$WCALC 784	Automatic Restart
%%\$WEEK 785	Restart 448
%%\$WEEKDAY 786	Automatic Tape Adjustment
%%\$YEARWK# 787	Description 55
%%CALCDATE 788	RESOURCE Parameter 597
%%SUBSTR 788	WM2744 Wish 55
AutoEdit Variables 747	Automatic Tape Adjustment Facility Statistics Screen 241
JCL Setup 788, 789	
AutoEdit Operator	Automation Log
%%MINUS 781	MODE Parameter 720
%%PLUS 747, 781	Automation Log Screen
Boolean Logic 773	SHOW Command 298
AutoEdit Parameter	AUTOMATION OPTIONS
OCCUR NO Suffix 835	IOA Menu 89
Parameter Prompting 337, 343, 347, 353, 358, 824, 829	AutoRefresh
AutoEdit Resolution	PA1 Key 95
Rerun/Restart Window 228	AutoRefresh Mode
AutoEdit SET Statement	Active Environment Screen 168
JCL SET 770	Screen Updating 100
AUTOEDIT SIMUL Option	View Graph Screen 201
Online Utilities Menu 323	AutoSave Documentation 146
AutoEdit Simulation	AUTOTAPE Parameter
CTMAESIM Utility 328	CTMPARM 241, 598
AutoEdit Statement	AUTOARCHIVE Parameter
CTMEXEC CLIST 835	Description 398
Syntax Checking 851	Example 400
AutoEdit Symbol	Average Statistics Line
%%JOBNAME 485	Statistics Screen 239
%%MEMNAME 485	
DO SYSOUT Statement 485	_
AutoEdit Syntax	В
Checking 328, 793	
CTMAESIM Utility 793	B Option
CTMSCIM Utility 333	Table List Screen 121
Testing 328, 793	Backslash Character
AutoEdit Variable	Global Variable 754
%%SIGN 599	Balancing Specifications
CMEM 701	DO CTBRULE Statement 442
Concatenation 767	BALANCING STATUS Option
Description 745	Primary Option Menu 89
DO MAIL Statement 453	BAPIACON Value
DO REMEDY 461	CTMBAPI AJF Action 896
DO SET Statement 465, 466, 795	BAPIACRT Value
DO SHOUT Statement 471, 631	CTMBAPI AJF Action 896
DO Statement 434	BAPIADEL Value
Linking 767	CTMBAPI AJF Action 896
MEMLIB Parameter 524, 796	BAPIAFOK Value
Multiple 767	CTMBAPI AJF Action 896
OVERLIB Parameter 572, 796	BAPIAFRE Value
O V EILLID I AIAIIICICI J (4. / 30	

CTMBAPI AJF Action 896	CTMBAPI Field 888
BAPIAHLD Value	BAPISIGN Field
CTMBAPI AJF Action 896	CTMAPI 887
BAPIAJID Field	BAPISJID Field
CTMBAPI 896	CTMBAPI 888
BAPIAMEM Field	BAPISJNM Field
CTMBAPI 896	CTMBAPI 888
BAPIAOID Field	BAPISLIB Field
CTMBAPI 896	CTMBAPI 888
BAPIARER Value	BAPISMEM Field
CTMBAPI AJF Action 896	CTMBAPI 888
BAPIAUND Value	BAPISODT Field
CTMBAPI AJF Action 896	CTMBAPI 888
BAPICMD Field 898	BAPISOID Field
CTMAPI 900	CTMBAPI 888
CTMBAPI 885	BAPISOWN Field
BAPICMDA Field	CTMBAPI 888
CTMAPI 885	BAPISRBA Field 888
BAPICMDL Field	BAPISRBB Field
CTMAPI 901	CTMBAPI 888
CTMBAPI 885	BAPISRNM Field
BAPIFLG1 Field	CTMBAPI 888
CTMBAPI 886	BAPISSTT Field
BAPIMCT Field	CTMBAPI 889
CTMBAPI 886	BAPISTAB Field
BAPIOWN Field	CTMBAPI 889
CTMBAPI 896	BAPISTAG Field
BAPIRBAC Field	CTMBAPI 889
CTMBAPI 896	BAPISTYP Field
BAPIRBAN Field	CTMBAPI 889
CTMBAPI 896	BAPIURC Field
BAPIRC Field	CTMAPI Return Codes 882
CTMAPI 886	CTMBAPI 887
CTMAPI Return Codes 882	BAPIVERS Field
BAPIRESP Quantitative Resource Input Field, CTMAPI	CTMAPI 887
899	BAPIWORK Field
BAPIRESQ Quantitative Resource Input Field, CTMAPI	CTMBAPI 887
899	Basic
	STAT CAL Parameter 639
BAPIRESX Quantitative Resource Input Field CTMAPI	
899	Basic Scheduling Parameters
BAPIRESX Quantitative Resource Input Field, CTMAPI	Group Entity 141
899	Scheduling Definition 131
BAPIRPL# Field	Summary 379
CTMAPI 886	Batch Job
BAPIRPLC Field	TASKTYPE Parameter 654
CTMBAPI 886	Batch Procedure
Initial Setting 903	CTMAESIM 328
BAPIRPLE Field	BLANK Value
CTMAPI 886	
	Parameter Prompting 354
BAPIRPLS Field	BLT Function
CTMAPI 886	CTMAPI, Setting Reply Fields 901
BAPIRSN Field	Reply Codes 901
CTMAPI Return Codes 882	BLT Function Replies
CTMBAPI 887	CTMAPI 901
BAPISGRN Field	BLT Function, CTMAPI
CTMBAPI 888	Procedure 900
RAPISHI D	BMC Software contacting 2

Boolean Logic	Inserting a New Year 313
Example 808	Overview 61
JCL Setup 772	Periodic Calendar 316
BOTTOM Command	Scheduling Jobs 52
Scrolling 96	CALENDAR Field
Bottom Line	Calendar Facility Entry Panel 309
Primary/Alternate 169	CALENDAR LIBRARY Parameter 328
Branching	Calendar List Screen 309
%%GOTO Control Statement 772, 808	Exiting 323
Browse Mode	Calendar Name
CMEM Rules 251	DCAL Parameter 421
DOCU/TEXT 145	Calendar Periodic
DOCU/TEXT Library 488, 490 Job List Screen 123	Description 316
	CANCEL Command
Zoom Screen 215	Calendar Definition Screen 321
BROWSE Option	CMEM Rule Definition 264
Calendar List Screen 310	Description 98
CMEM Table List 255	Scheduling Definition 148
Table List Screen 114, 121	Scheduling Definition Entry Panel 118
BUT NOT FOUND n TIMES	Zoom Screen 221
Active Environment Screen 185	CAPS Command
BYPASS Option	CMEM Rule Definition 262
Active Environment Screen 184	Variable Zoom screen 280
Default Settings 184	CAPS OFF
Fields 184	Command 262
Bypassing	CAPS ON
Table List Screen 264	Command 262
	CATEGORY Command
	Log Screen 298
	CATEGORY Field
	Job Scheduling Table 814
C Option	CATEGORY Parameter 380
Active Environment Screen 182	CCTMJOB
Job List Screen 126, 257	Replacement by CTMAPI 864
Year List Screen 313	CHANGE Command
Cache	Scheduling Definition 143
%%GLOBAL Members 758	
	CHANGE Option
Calendar	CHANGE Option Conditions/Resources 287, 288
	Conditions/Resources 287, 288
DATES Parameter 418	Conditions/Resources 287, 288 CHANGE RESOURCE Parameter Simulation 845
DATES Parameter 418 DAYS Parameter 422	Conditions/Resources 287, 288 CHANGE RESOURCE Parameter Simulation 845 Character
DATES Parameter 418 DAYS Parameter 422 Example 424, 673	Conditions/Resources 287, 288 CHANGE RESOURCE Parameter Simulation 845 Character Global Variable 754
DATES Parameter 418 DAYS Parameter 422 Example 424, 673 Job Scheduling Plan 165	Conditions/Resources 287, 288 CHANGE RESOURCE Parameter Simulation 845 Character Global Variable 754 Gregorian Date Format 66
DATES Parameter 418 DAYS Parameter 422 Example 424, 673 Job Scheduling Plan 165 Periodic 53	Conditions/Resources 287, 288 CHANGE RESOURCE Parameter Simulation 845 Character Global Variable 754 Gregorian Date Format 66 Hex Value 82
DATES Parameter 418 DAYS Parameter 422 Example 424, 673 Job Scheduling Plan 165 Periodic 53 Regular 53	Conditions/Resources 287, 288 CHANGE RESOURCE Parameter Simulation 845 Character Global Variable 754 Gregorian Date Format 66 Hex Value 82 Character Masking 83
DATES Parameter 418 DAYS Parameter 422 Example 424, 673 Job Scheduling Plan 165 Periodic 53 Regular 53 WDAYS Parameter 670	Conditions/Resources 287, 288 CHANGE RESOURCE Parameter Simulation 845 Character Global Variable 754 Gregorian Date Format 66 Hex Value 82 Character Masking 83 CHECK IN EXT VOL Option
DATES Parameter 418 DAYS Parameter 422 Example 424, 673 Job Scheduling Plan 165 Periodic 53 Regular 53 WDAYS Parameter 670 Calendar Date	Conditions/Resources 287, 288 CHANGE RESOURCE Parameter Simulation 845 Character Global Variable 754 Gregorian Date Format 66 Hex Value 82 Character Masking 83 CHECK IN EXT VOL Option Primary Option Menu 89
DATES Parameter 418 DAYS Parameter 422 Example 424, 673 Job Scheduling Plan 165 Periodic 53 Regular 53 WDAYS Parameter 670 Calendar Date System Variable 750	Conditions/Resources 287, 288 CHANGE RESOURCE Parameter Simulation 845 Character Global Variable 754 Gregorian Date Format 66 Hex Value 82 Character Masking 83 CHECK IN EXT VOL Option Primary Option Menu 89 CICS 81
DATES Parameter 418 DAYS Parameter 422 Example 424, 673 Job Scheduling Plan 165 Periodic 53 Regular 53 WDAYS Parameter 670 Calendar Date System Variable 750 CALENDAR DEF Option	Conditions/Resources 287, 288 CHANGE RESOURCE Parameter Simulation 845 Character Global Variable 754 Gregorian Date Format 66 Hex Value 82 Character Masking 83 CHECK IN EXT VOL Option Primary Option Menu 89 CICS 81 CMEM On Spool Job 683
DATES Parameter 418 DAYS Parameter 422 Example 424, 673 Job Scheduling Plan 165 Periodic 53 Regular 53 WDAYS Parameter 670 Calendar Date System Variable 750 CALENDAR DEF Option Primary Option Menu 86	Conditions/Resources 287, 288 CHANGE RESOURCE Parameter Simulation 845 Character Global Variable 754 Gregorian Date Format 66 Hex Value 82 Character Masking 83 CHECK IN EXT VOL Option Primary Option Menu 89 CICS 81 CMEM On Spool Job 683 Environment 577
DATES Parameter 418 DAYS Parameter 422 Example 424, 673 Job Scheduling Plan 165 Periodic 53 Regular 53 WDAYS Parameter 670 Calendar Date System Variable 750 CALENDAR DEF Option Primary Option Menu 86 Calendar Definition Screen 311, 315	Conditions/Resources 287, 288 CHANGE RESOURCE Parameter Simulation 845 Character Global Variable 754 Gregorian Date Format 66 Hex Value 82 Character Masking 83 CHECK IN EXT VOL Option Primary Option Menu 89 CICS 81 CMEM On Spool Job 683 Environment 577 OWNER Parameter 577
DATES Parameter 418 DAYS Parameter 422 Example 424, 673 Job Scheduling Plan 165 Periodic 53 Regular 53 WDAYS Parameter 670 Calendar Date System Variable 750 CALENDAR DEF Option Primary Option Menu 86 Calendar Definition Screen 311, 315 Exiting 320, 321	Conditions/Resources 287, 288 CHANGE RESOURCE Parameter Simulation 845 Character Global Variable 754 Gregorian Date Format 66 Hex Value 82 Character Masking 83 CHECK IN EXT VOL Option Primary Option Menu 89 CICS 81 CMEM On Spool Job 683 Environment 577 OWNER Parameter 577 PF06/PF18 93
DATES Parameter 418 DAYS Parameter 422 Example 424, 673 Job Scheduling Plan 165 Periodic 53 Regular 53 WDAYS Parameter 670 Calendar Date System Variable 750 CALENDAR DEF Option Primary Option Menu 86 Calendar Definition Screen 311, 315 Exiting 320, 321 Calendar Facility	Conditions/Resources 287, 288 CHANGE RESOURCE Parameter Simulation 845 Character Global Variable 754 Gregorian Date Format 66 Hex Value 82 Character Masking 83 CHECK IN EXT VOL Option Primary Option Menu 89 CICS 81 CMEM On Spool Job 683 Environment 577 OWNER Parameter 577 PF06/PF18 93 CICS Environment 81
DATES Parameter 418 DAYS Parameter 422 Example 424, 673 Job Scheduling Plan 165 Periodic 53 Regular 53 WDAYS Parameter 670 Calendar Date System Variable 750 CALENDAR DEF Option Primary Option Menu 86 Calendar Definition Screen 311, 315 Exiting 320, 321 Calendar Facility Accessing 307	Conditions/Resources 287, 288 CHANGE RESOURCE Parameter Simulation 845 Character Global Variable 754 Gregorian Date Format 66 Hex Value 82 Character Masking 83 CHECK IN EXT VOL Option Primary Option Menu 89 CICS 81 CMEM On Spool Job 683 Environment 577 OWNER Parameter 577 PF06/PF18 93 CICS Environment 81 Class
DATES Parameter 418 DAYS Parameter 422 Example 424, 673 Job Scheduling Plan 165 Periodic 53 Regular 53 WDAYS Parameter 670 Calendar Date System Variable 750 CALENDAR DEF Option Primary Option Menu 86 Calendar Definition Screen 311, 315 Exiting 320, 321 Calendar Facility Accessing 307 Deleting a Calendar 319	Conditions/Resources 287, 288 CHANGE RESOURCE Parameter Simulation 845 Character Global Variable 754 Gregorian Date Format 66 Hex Value 82 Character Masking 83 CHECK IN EXT VOL Option Primary Option Menu 89 CICS 81 CMEM On Spool Job 683 Environment 577 OWNER Parameter 577 PF06/PF18 93 CICS Environment 81
DATES Parameter 418 DAYS Parameter 422 Example 424, 673 Job Scheduling Plan 165 Periodic 53 Regular 53 WDAYS Parameter 670 Calendar Date System Variable 750 CALENDAR DEF Option Primary Option Menu 86 Calendar Definition Screen 311, 315 Exiting 320, 321 Calendar Facility Accessing 307 Deleting a Calendar 319 Entry Panel 308	Conditions/Resources 287, 288 CHANGE RESOURCE Parameter Simulation 845 Character Global Variable 754 Gregorian Date Format 66 Hex Value 82 Character Masking 83 CHECK IN EXT VOL Option Primary Option Menu 89 CICS 81 CMEM On Spool Job 683 Environment 577 OWNER Parameter 577 PF06/PF18 93 CICS Environment 81 Class
DATES Parameter 418 DAYS Parameter 422 Example 424, 673 Job Scheduling Plan 165 Periodic 53 Regular 53 WDAYS Parameter 670 Calendar Date System Variable 750 CALENDAR DEF Option Primary Option Menu 86 Calendar Definition Screen 311, 315 Exiting 320, 321 Calendar Facility Accessing 307 Deleting a Calendar 319	Conditions/Resources 287, 288 CHANGE RESOURCE Parameter Simulation 845 Character Global Variable 754 Gregorian Date Format 66 Hex Value 82 Character Masking 83 CHECK IN EXT VOL Option Primary Option Menu 89 CICS 81 CMEM On Spool Job 683 Environment 577 OWNER Parameter 577 PF06/PF18 93 CICS Environment 81 Class Allocation 804, 805

CLEANUP Option	Rule Management 682
Online Utilities Menu 323	CMEM Message Type
CLEANUP Status	IOA Log Show Screen Window 303
Active Environment Screen 185	CMEM Monitor
CLIST	Variable Database Facility 271
Activation 109	CMEM On Spool Job
TSO Environment 813	On Spool Job 705
CLIST CTMCAES	CMEM Option
AutoEdit Syntax Testing 328	Primary Option Menu 250, 252
CLIST CTMCAMNU	CMEM Rule
Parameter Prompting 350	And/Or/Not Logic 722
CLIST CTMCDOCU	Browsing 251
DOCU/TEXT Product 372	Comments 262
CLIST CTMCFMNU	Components 684
Parameter Prompting 340	CONTROL-O Rule 695
CLIST CTMCSIM	Creation 251
Simulation/Tape Pull 332	Dataset Event 680, 724
CLIST CTMEXEC	Definition 693
Parameter Prompting 832, 834	DESCRIPTION Parameter 697
CLIST CTMFETCH	DO COND Statement 681, 700
Parameter Prompting 832	DO FORCEJOB Statement 681, 703
CLIST CTMJBINT	DO PORCESOB statement 001, 703  DO RULE Statement 707
End User Job Order 370	DO SHOUT Statement 710
Job Ordering 813 CLIST CTMJOBRQ	DO Statement 681, 695, 699
	DO STOPJOB Statement 681, 715
Job Ordering 326, 813	Editing 252
CLIST CTMPROMPT	GROUP Parameter 717
Quick Schedule Definition 360	Job Arrival 729
CLIST CTMQUICK	Job Scheduling Definition 685
Quick Schedule Definition 361	Management 681
CLIST IOAUTIL	MODE Parameter 719
Online Utilities 323	ON DSNEVENT Statement 680, 682, 684, 721, 724
CM VER Parameter	ON JOBARRIV Statement 680, 684, 721, 729
Description 401	ON JOBEND Statement 680, 721, 731
CM Version	On Spool Job 683
CM VER Parameter 401	ON Statement 721
CMEM	ON STEP Statement 680, 684, 721, 733
FTP products 691	OWNER Parameter 737
Generation Data Sets 691	Parameters 693
IBM FTP 692	Prerequisite Condition 681, 700
SMS Support 691	Screen 693, 697, 714, 716, 720, 728, 730, 732, 740
CMEM DEFINITION Option	Simulation 719
IOA Primary Option Menu 87	CMEM Rule Definition
CMEM Entry Panel	Commands 261
Exiting 266	Description 258
CMEM Facility	Editing 263
Actions 681	Entry Panel 253
AutoEdit Variables 701	Exiting 264
CICS Job 683	CMEM Rule Facility
CONTROL-O 52	Description 249
Description 680	Exiting 264
DO FORCEJOB Statement 704	ISPF PACK Option 250
Event Types 52, 680	Screens 250
External Events 51	TEST Mode 719
Forced Job 686, 704	CMEM Rule List
On Spool Job 52, 683	Browse Mode 255
Overview 60	Exiting 264

Options 257	AUTO 100, 176
Screen 254, 697	BOTTOM 96
CMEM Rule Table	CANCEL 99, 118, 148, 221, 264
Creation 250	Change Color 163
Deletion 266	CMEM Rule Definition 261
List 254	Conditions/Resources Screen 286
Ordering 267, 682	Copy 915, 928
CMEM Security	CPUID 178
RUNTSEC Parameter 740	CTMQSB 813
CMEM support	CTMTTRA 109
ON STEP Statement 688	DATA 124
CMEM Table	
	Delete 915, 928
Ordering 252	DESC 124, 178, 256, 312
CMEM Table List	DISPLAY 169, 174
Exiting 266	DOC 145
Options 254	DOWN 93
Statistics 254	DUMP 179
Cnnnn Code	EDIT 143, 262, 263, 913
+EVERY Step 549	END 93, 148, 200, 264, 306
Code *****	Exit Online Facility 91
FLUSH Code 554	FIND 93, 96, 162
SNRUN Code 554	GROUP 179, 298
CODE Criteria	HELP 93, 99
IOA Log Show Screen Window 303	HISTORY 175
Screen Filter 303	Insert 916, 929
CODE Field	IOA Primary Option Menu 86
Log Screen 297	ISPF 101
CODES Parameter	JCL 115
ON Statement 544, 550	JES HOLD 186
Collection	Job Dependency 244
MAINVIEW Batch Optimizer (MVBO)	Job Dependency Network 244
Implementation 822	JOBSTART 240
Color Support	JOBSTAT 142, 176
Active Environment Screen 168	KEYS 101, 238
Box Color 163	L Parameter Prefix 101, 343, 346, 354, 359
Graphic Jobflow 162	Line Editing 915, 928
Job Graph 203	LOCATE 96, 234
Online Facility 82	Location 916, 930
View Graph Screen 202	
Column Range	Log Screen 297 Move 916, 929
%%RANGE Control Statement 777	NEW COND 293
	NEW LCOND 293
COM-PLETE 81	NOTE 177
Combinatorial Logic	
CMEM ON Statement 722	OIDL 179
Command 1	Online Facility 93
IOA Primary Option Menu 87, 88	OPT 174
Command Line	ORDERID 180
Online Facility 92	PRINT 101
Command line commands	Quick Submit 813
IOA Editor 104	RBAL 175
Commands	REFRESH 100, 176, 200, 242, 244
=6 93, 109	Repeat 916, 929
Active Environment Screen 174	RESET 93, 98, 99, 150, 199, 265, 306, 369, 913
ADD COND 286	RETRIEVE 94
ADD CONTROL 286	Rule Editing 927
ADD RESOURCE 286	SAVE 219, 221
ADD Resources 285	Scheduling Definition 142

Scrolling 95	condition name
SELECT 112, 122	AutoEdit variable in 294, 437, 500, 562, 796
SET 105	CONDITION NAME Field
SHOW 93, 175, 190, 300	Prerequisite Condition Utility 325
SHPF 94	Condition Names
SORT 125	Long 436, 502, 565
SPLIT 101	Condition Parameter
STAT 124, 257, 312	DO COND Statement 700
SWAP 101	CONDITION/RESOURCE Field
Sysout Viewing 237	Conditions/Resources 283
TABLE 177	Conditional Processing
Table List Screen 122	DO Statement 386
TOP 96	IF, THEN, ELSE 772, 809
TSO 108	ON Statement 386
TSO CTMTTRA 109	Conditions
UP 93, 95	Forecasting 840
Utilities Transfer 93	Simulation 840
VG 180, 201	Conditions File 62
VIEW 180, 199	Conditions/Resources
VIEW GRAPH 180, 201	Add/Check/Delete Utility 324
VIEWALL 236	Delete Option 287
Comment	Fields 282, 290
CMEM Rule Definition 262	Handling 48, 73
Comparison Operators	IOALDNRS Utility 817
AutoEdit Logic 773	Manual Conditions 294
Components	Options 287
CONTROL-M 44	Restoration 49
Compression Job	condopt Parameter
CONTROL Parameter 410	DO COND Statement 701
	CONFCAL Calendar
Computer Allocation	
Example 805	Schedule Validation 381
Computer ID	Scheduling Logic 382
Started Task 526	CONFCAL Parameter
COMPLETE 81	Description 402
PF06/PF18 93	MINIMUM Parameter 531
Concatenation	PDS Parameter 579
%% Symbol 748, 769	Periodic Calendar 404
AutoEdit Variable 767	Configuration Table
Logic 767	AutoEdit Statement 836
COND Field	CONFIRM Field
Conditions/Resources 284	CMEM Rule Table Order 268
COND-NAME Subparameter	Conditions/Resources 288
DO COND Parameter 437	Manual Conditions 295
IN Parameter 500	Restart Confirmation Window 225
OUT Parameter 562	Scheduling Confirmation 153
COND/RES Option	Zoom Screen 216
Primary Option Menu 86	CONFIRM Option
Condition Code	Active Environment Screen 182
ON Statement 550	CONFIRM Parameter
Condition Code Recapture	Description 406
Restart Confirmation 226	DO IFRERUN Statement 449
CONDITION DATE Field	Example 407
Prerequisite Condition Utility 326	Confirm Rerun Window
CONDITION Field	Active Environment Screen 222, 223
Parameter Prompting 343	Confirm Restart Window
CONDITION Name	Active Environment Screen 224
Manual Conditions 292	Confirm Scheduling Window

Active Environment Screen 222	Restart under, Job Status 189
Confirmation Window	CONTROL-O
CMEM Rule Table 268	Product Description 43
DO IFRERUN Statement 450	SHOUT Facility 52
Force OK 247	CONTROL-O Monitor
Manual Conditions 294	Variable Database Facility 271
Manual Scheduling 151	CONTROL-O/COSMOS
Why Screen 208	Product Description 43
Zoom Screen 221	CONTROL-D/Image
CONNECT DIRECT	Product Description 43
File Transfer 702	CONTROL-M
CONTROL Field	Concepts 61
Conditions/Resources 284	Overview 43
CONTROL Parameter	Product Description 42
Description 408	CONTROL-M Repository 62
Example 410	CONTROL-O
Logic 410	%%\$ARGS System Variable 708
CONTROL Resource	Automation Log 719
Adding 285	CMEM Facility 52
Critical Path Priority 589	CMEM Rule 681
Definition 73	MODE Parameter 719
Manual Addition 286	Shout Facility 695
Runtime Criteria 48	CONTROL-O Rule
Show Screen Filter 197	DO RULE Statement 707
CONTROL Resources	CONTROL-V
CONTROL-M Resources File 281	Product Description 43
IOA Conditions/Resources Screen 281	CONTROL-D
CONTROL Statements	D-CAT Parameter 416
AutoEdit 770	Product Description 43
CONTROL-M	CONTROL-D/Page On Demand
Application Program Interface 862	Product Description 43
Implementation 812	CONTROL-M Monitor
Parameter Prompting 340	Description 45
SIMPARM DD Statement 843	Postprocessing 50
CONTROL-M Monitor	CONTROL-M Status Field
Multiple Monitors 55	Active Environment Screen 171
New functions 572	CONTROL-M/Tape
Simulation Facility 333	Product Description 43
CONTROL-M Resources	CONTROL-M/Restart
File 281	DO IFRERUN Statement 447
CONTROL-M Resources File 62, 282	Product Description 42
CONTROL M Resources The 02, 202	Restart under 189, 447
CTB STEP Parameter 413	Restart Window 224
DO CTBRULE Statement 442	SNRUN Code on Restart 554
Product Description 43	CONTROL-O
CONTROL-M/Enterprise Manager	CMEM Rule 695
APPL Parameter 393	DO SHOUT Statement 710
DO SHOUT Destination 471, 711	Rule Invocation 681
GROUP Parameter 496	Shout Facility 695
SHOUT Parameter 629	Conventions Used in This Guide 34
CONTROL-M/Restart SIMUL Option	Conversational Mode
Online Utilities Menu 323	CTMAPI 882
CONTROL-M/Restart	COPMEM2O Parameter 572
FLUSH Code on Restart 553	Copy Commands
History Jobs File 49	Edit Environment 915, 928
Rerun Confirmation 225	COPY Option
Restart under 67	Job List Screen 115

Year List 313	AJF Action, BAPIAFOK Value 896
Copy Option	AJF Action, BAPIAFRE Value 896
Job List Screen 126, 257	AJF Action, BAPIAHLD Value 896
Copying	AJF Action, BAPIAKIL Value 896
Jobs 157	AJF Action, BAPIARCT Value 896
Sysout 478, 482, 646, 649	AJF Action, BAPIARER Value 896
Copying Jobs 156	AJF Action, BAPIAUND Value 896
Copying Rules 269	Allocations 863
COSMOS Status Option	Available Functions 864
Primary Option Menu 89	BAPICMD Field 885, 900
COUNT Parameter	BAPICMT Field 886
Change Resource Window 289	BAPIGOPT Global Variable Input Field 898
CPU Field	BAPIRC Field 886
Active Environment Screen 172	BAPIRESN Quantitative Resource Input Field 899
CPU ID	BAPIRESP Quantitative Resource Input Field 899
Version Information Window 90	BAPIRESQ Quantitative Resource Input Field 899
CPU Time Field	BAPIRESX Quantitative Resource Input Field 899
Statistics Screen 240	BAPIRPL Field 886
CPU Time, Group	BAPIRPL# Field 903
Statistics Screen 240	BAPIRPLC Field 903
cpuid	BAPIRPLC Field, Initial Setting 903
MEMLIB Parameter 524	BAPIRPLE Field 886, 903
CPUID Command	BAPIRSN Field 887
Active Environment Screen 178	BAPISIGN Field 887
CPUID Field	BAPIURC Field 887
Zoom Screen 217	BAPIVERS Field 887
CREATE Field	BLT Function 900
Exit Option Window 150	BLT Function Reply Codes 901
Criteria 47	BLT Function, procedure 900
Critical Path	
	BLT Function, Setting Reply Fields 901
Deleting a Job 211	Calling 862
Job Dependency 74 PRIORITY Parameter 588	Calling AJF Actions 870 Causes of failure 882
Resource Allocation 74	CLIST and 862
Cross Memory Interface	Conditional Requests 879
Online Monitor 81	Conditional Selection Criteria 879
CRSR	Contents of Input and Output Registers 883
Scrolling Amount 96	Conversational Mode 863, 882
CST Task Type	Create New Tables 868
IOA Log Show Screen Window 304	CTMBAPI Mode 902
Show Screen Filter 196	CTMBLT Replacement 868
CTB STEP Parameter	DAAPI DD Statement 862
Description 413	Date Field Format 903
Example 414	Environment 863
CTGFORC Parameter	Environments 862
CTMPARM 417	Fields in the Fixed (Header) Part 884
CTMAESIM Utility	Fixed Part Values 885
AutoEdit Syntax 328, 793	Force Jobs Using 865
CTMAJO	Force New Tables 868
Replaced by CTMBLT Utility 815	Force Return Codes 867
CTMAJO Program	Forcing Allocations 867
Job Ordering 815	Forcing under CLIST 866
Work Flow 816	Forcing under REXX 866
CTMAPI	Forcing Using a Program 867
AJF Action Return Codes 897	Generally 862
AJF Action, BAPIACON Value 896	Global Variable Return Codes 899
AJF Action, BAPIADEL Value 896	Invoking Search from a Program 873

IOA Variables Database and 865	BAPIRPLS Field 886
Masking in Character Fields 879	BAPISGRN Field 888
Multiple IF Statements 880	BAPISHLD Field 888
Order Extension 892	BAPISJID Field 888
Order Jobs Using 865	BAPISJNM Field 888
Order New Tables 868	BAPISLIB Field 888
Order Return Codes 867	BAPISMEM Field 888
Ordering Allocations 867	BAPISODT Field 888
Ordering under CLIST 866	BAPISOID Field 888
Ordering under REXX 866	BAPISOWN Field 888
Ordering using a Program 867	BAPISRBA Field 888
Pre-allocating SYSPRINT 866	BAPISRBB Field 888
Quantitative Resource Extension Function 899	BAPISRNM Field 888
Quantitative Resource Input Fields 899	BAPISSTT Field 889
Quantitative Resource Return Codes 900	BAPISTAB Field 889
Reply Mechanism Trigger Pointers 903	BAPISTAG Field 889
Requirements before Calling 863	BAPISTYP Field 889
REXX and 862	BAPIWORK Field 887
Scanning and Filtering Reply Lines 903	Components 884
Search Call Syntax 872	DSECT 884
Search Function 872	Replies 902
Security Exit IOASE07 900	Status Extension Fields 888
Selection Criteria and Performance 880	Status Reply DSECT 889
Selection Criteria Parameter Attributes 881	Statuses Returned 890
Status Function 887	CTMBAPI DESCT 863
Syntax for executing 867, 871	CTMBJSE
Syntax Forcing Jobs 865	DESCT 889
Syntax of AJF Action under 870	CTMBLT
Syntax of Search Call 872	Parameters 869
Syntax Ordering Jobs 865	Replacement by CTMAPI 864, 868
Tailoring 863	CTMBLT Utility
Under IOA Environment 864	Assembler Macro 377
Use to checkpoint variables 865	CTMAPI BLT Function and 900
Use to resolve variables 865	Job Ordering 813
Use to Search and Query Job Details 865	Replacing CTMAJO 815
Use to Search and Query Job Status 865	CTMCAES CLIST
Use to set variables 865	CTMAESIM Utility 793
CTMAPI DSECT 884	TSO Command 328
CTMAPI Replies	CTMCAES Option
Reply Mechanism Trigger Pointers 903	CTMAESIM Utility 793
CTMAPI Return Codes	CTMCAES Utility
BAPIRC Field 882	AutoEdit Simulation 324
BAPIRSN Field 882	CTMCAJF Utility
BAPIURC Field 882	AUTOARCHIVE Parameter 399
Generally 882	CTMCAMNU Option
CTMBAPI	Parameter Prompting Entry Panel 33'
BAPIAGRN Field 896	CTMCFMNU Option
BAPIAMEM Field 896	Parameter Prompting Entry Panel 33'
BAPICMDA Field 885	CTMCSIM CLIST
BAPICMDL Field 885	TSO Command 332
BAPIFLG1 Field 886	CTMEXEC CLIST
BAPIJID Field 896	Example 835
BAPIJNM Field 896	Parameter Prompting 832
BAPIOWN Field 896	CTMEXEC Option
BAPIRBAC Field 896	Parameter Planning 356
BAPIRBAN Field 896	Parameter Printing 350
BAPIRPLC Field 886	CTMFETCH CLIST
D. II IIVI LO I ICIU VVV	

Parameter Prompting 832	CTMQSB Command 814
CTMFETCH Option	CTMX002 Exit
Parameter Prompting 350, 355	CTMAESIM Utility 793
CTMJBINT CLIST	MEMLIB Parameter 525
TSO Command 370	Parameter Prompting 837
CTMJBINT Utility	RESOURCE Parameter 599
End User Job Order 370	Simulation 847
Job Order Interface 324	CTMX003 Exit
Job Ordering 813	Simulation 847
CTMJOB Utility	CTMX004 Exit
Job Ordering 812, 813	RESOURCE Parameter 599
TSO environment 813	Scheduling Algorithm 601
CTMJOBRQ CLIST	CTMX010 User Exit
TSO Command 326	CTMQSB Command 814
CTMJOBRQ Utility 324	CTMX013 Exit
Job Order Request 326	Statistics Screen 240
Job Ordering 813	CTMX015C Exit
CTMJSA Utility	Functions 572
Statistics File Update 238	CTMX015O Exit
CTMPARM	Functions 572
#JNFRT Parameter 217	CTMX2PPF Member
AECACHL Parameter 758	IOA SAMPEXIT Library 837
AUTOTAPE Parameter 241, 598	CTO147I message 692
CTGFORC Parameter 417	CTO282I Subparameter
DUEINCHK Parameter 493	DO SHOUT Statement 711
FRCOKOPT Parameter 247, 553	CTO403I pseudo-message 692
GRPRECHK Parameter 112	CTO782I message 692
MAXCCOK Parameter 552, 553	CTO783I message 692
OVERJCLM Parameter 536, 652	customer support 3
OVERJCLM parameter 619	Customization
TAGMAXWT Parameter 520	IOA 80
CTMPLEX	Customizing
Minus-One Support 55	Options 58
CTMPROMP Utility 324	CYC Task Type
CTMQSB Command	IOA Log Show Screen Window 304
CTMX010 Exit 814	Show Screen Filter 196
Job Ordering 813	Cyclic Job
CTMQSC Application	AUTOARCHIVE Parameter 399
CTMAJO Program 815	CONFIRM Parameter 406
CTMQUICK CLIST	INTERVAL Parameter 513
TSO Command 360	TASKTYPE Parameter 654
CTMQUICK Option	Cyclic Jobs
Online Utilities Menu 361	Force OK 247
CTMQUICK Utility	Cyclic Jobs Stopping 475
Example 369	Cyclic Started Task
Quick Schedule Definition 360	TASKTYPE Parameter 654
Schedule Definition 324	Thomas a summer of the
Tables 113	
CTMRSTR Utility	D
Restoration 49	
CTMSIM Procedure	D JOB Message Type
Simulation Procedure 840	IOA Log Show Screen Window 303
CTMTAPUL Procedure	D Option
Tape Pull List 840, 851, 853	Active Environment Screen 181
CTMWORK Value	Job List Screen 125
SYSOPT Variable 443	Parameter Prompting 344
CTMX001 Exit	Table List Screen 122
CIMAUUI EAIL	

D-CAT Field	DASIMPRM DD Statement
Ignored by CTMQSB Command 814	Simulation Parameter 843
D-CAT Parameter	DASTAT DD Statement
Category E 380	Simulation Statistics 845
Description 416	DASUBMIT DD Statement
Example 417	AutoEdit Simulation 331
D-INT Message Type	CTMAESIM Utility 794
IOA Log Show Screen Window 303	Emergency Execution 794
DAACTLOG DD Statement	Data Area of Screen
MODE Parameter 719	Online Facility 92
DACKPTIN DD Statement	DATA Command
Simulation Active Jobs File 842	Job List Screen 124
Tape Pull List 854	DATA Format
DACKPTOU DD Statement	Job List Screen 124
Simulation Active Jobs File 846	Database Facility 271
DACNDF DD Statement	Database List Screen
CONTROL-M Resources Simulation 842	Variable Database Facility 273
Simulation 850	Database Update
DAGLBLST DD	IN Parameter 508
Variable Database Facility 273	DATAPNAM DD Statement
DAGLOBAL Statement 758	Tape Pull List 855
%%GLOBAL Control Statement 771	dataset cleanup 585
PARAMETER LIBRARY Parameter 794	Dataset Disposition
Daily AutoEdit Member	ON DSNEVENT Statement 724
Parameter Prompting 831	DATASET Event
Daily JCL Library	CMEM 680
Allocation 838	DO FORCEJOB Statement 704
Deletion 838	Dataset Event
Parameter Prompting 831	ON DSNEVENT 680, 682, 721
Daily Plan	ON DSNEVENT Statement 725
Parameter Prompting 357, 831	Dataset Name
Daily Prompting Table	CMEM 682
Daily Table 345	Date Calculation
Daily Scheduling Table	%%\$CALCDTE Function 801
Parameter Prompting 830	Date Definition
Daily Subsystem	Overview 63
D-CAT Parameter 416	DATE Field
DCAL Calendar 421	Conditions/Resources 284
Daily Table	Job Order Execution History 236
Table Selection Screen 345, 827	Log Screen 297
DAJCLOUT DD Statement 854	Simulation/Tape Pull 334
JOB/SCAN DOCU/TEXT 855	Date Field
Tape Pull List 854, 855	Why Screen Confirmation 209
DALIB DD Statement	Date Field Format
MEMLIB Parameter 523	CTMAPI 903
DALOGIN DD Statement	DATE Parameter
Tape Pull List 854	CTMEXEC CLIST 834
DALOGOUT DD Statement	CTMFETCH CLIST 833
Simulation Log File 846	DO FORCEJOB Statement 703
DANRES DD Statement	DATE Range
Simulation 850, 851	Job Scheduling Plan Screen 166
DANSINC DD Statement	Log Screen 297
Simulation 850, 851	Manual Conditions 293
DAREPOUT DD Statement	Date Range
Tape Pull List 855	Log Screen 297
DASIMOUT DD Statement	DATE Reference
Simulation Messages 846	Manual Conditions 292

Date Reference	Primary Option Menu 89
DO COND Statement 437, 700	DB/2 database
Generic 508	use with SAP/R3 909
IN Parameter 501, 510	DCAL Parameter
January 1st 72, 700	Calendar Name 421
OUT Parameter 563, 570	Calendar Type 423
Prerequisite Condition 71	DATES Parameter 418
STAT 72, 437, 700	DAYS Parameter 381
Zoom Screen 218	MAXWAIT Example 522
Date Updated Field	Non-periodic Calendar 422
Parameter Prompting 347	DCAL parameter
Date Variable	Periodic Calendar 423
Example 797	DD Statement
JCL Setup 750	DAACTLOG 719
DATEMEM Calendar	DACKPTIN 842, 855
WCAL Parameter 670	DACKPTOU 846
dateref Parameter	DACNDF 842
DO COND Statement 700	DAGLBLST 273
DATEREF Subparameter	DAGLOBAL 758
DO COND Parameter 437	DAJCLOUT 855
IN Parameter 501	DALIB 523
OUT Parameter 563	DALOGIN 855
DATES Parameter	DALOGOUT 846
DAYS Parameter 424	DAREPOUT 855
Description 418	DASIMOUT 846
Example 419	DASIMPRM 843
MINIMUM Parameter 531	DASTAT 845
MONTHS Parameter 418, 533	DASUBMIT 332, 794
PDS Parameter 579	DATAPNAM 855
DATES Range Field	TAPULIN 853, 855
Conditions/Resources 285	TAPULOUT 855
DATETYP Parameter 430, 616	Deadline Adjustment
Day of the Week	Job Flow 75
First 669	DEADLINE Argument
WDAYS Parameter 669	REFRESH Command 245
DAYJCLB Parameter	Decollating Mission
CTMFETCH CLIST 833	D-CAT Parameter 416
DAYS Parameter	Default Display Type
DATES Parameter 418	Active Environment Screen 170
DCAL Field 381	Job Dependency Network 243
Description 420, 659	Job Order Execution History 235
Example 424	DEFAULT Field
Format 420, 421	Parameter Prompting 354
FROM DAYS 659	DEFAULT Filter
Logic 423	Active Environment Screen 175
MINIMUM Parameter 531	Default Filter
Negative Value 423	Active Environment Screen 190
PDS Parameter 579	Log Screen 299
Runtime 659	DEFAULT STATUS Field
Scheduling Logic 382	Parameter Prompting 359
UNTIL DAYS 659	Define Parameters and Conditions
DAYTBLB Parameter	Exiting 344
CTMEXEC CLIST 834	Fields 343
CTMFETCH CLIST 833	Options 344
DAYTIMEM Parameter	Screen 342
ODATE 64	Type 1 Option 1 341
DB VARIABLE DEF Option	Define Parameters in Master Plan

Fields 353	Active Environment Screen 178
Options 354	Job List Screen 124
Screen 352	Rule List Screen 256
DEFINITION ACTIVE Parameter	Year List Screen 312
Description 430	DESC Field
Forced Jobs 430	Active Environment Screen 172
Format 430	IOA Log Show Screen Window 302
FROM 133, 430	Parameter Prompting 343
UNTIL 133, 430	Show Screen Filter Window 193
DEL Option	DESC Format
Active Environment Screen 181	Job List Screen 124
DEL Status	DESC Parameter
CTMAPI 890	Description 432
Delete Commands	Example 432
Edit Environment 915, 928	Description
DELETE COND Parameter	THRESHOLD Parameter 741
Simulation Parameter 844	DESCRIPTION Field
Delete Confirmation Window	Quick Schedule Definition 367
Active Environment Screen 211	Rule List Screen 255
Table List Screen 158	Description Field
Delete NOT-COND Option	Parameter Prompting 347
Why Screen 208	DESCRIPTION Parameter
DELETE Option	CMEM Rule 697
Calendar List Screen 310	Example 697
CMEM Rule List 257	Scheduling Definition 126
CMEM Table List 255, 266	Destination Destination
Conditions/Resources 287	DO MAIL Statement 452
Job List Screen 115, 125	DO SHOUT Statement 472, 473, 633, 710, 713
Parameter Prompting 344, 354	DO SYSOUT Statement 477
Table List Screen 115, 122	SYSOUT Parameter 644
Year List Screen 312	Devices
Delete Option Table List Screen 158	Tape Usage 241 DEVICES USED Field
DELETED Status	Statistics Screen 241
Active Environment Screen 185	DISAPPEARED Status
Deleting Calendars 319	Activate Option 181 Active Environment Screen 185
CMEM Table 266 DO Statement 435	Show Screen Filter 195 Zoom Screen 217
	DISP Field
Manual Conditions 294	
MSGCLASS Sysout 484	WHY Option 290
ON Statements 546	DISP Parameter
Prerequisite Condition 72, 438	ON DSNEVENT Statement 725
Sysout 479, 482, 646, 649	DISPLAY Command
Table in Table List 266	IOA Log Screen 298
Deleting a Job	Job Order Execution History Screen 235
Group Entity 211	Status Command 174
WAIT SCHEDULE jobs 210	Variable Zoom screen 279
DELOVRER Parameter 572	Display Command
DELOVRUN Parameter 572	Active Environment Screen 169
Dependency	Display Filters Window
Maybe Job 819	Fields 301
DEPENDS ON Field	Options 191, 301
Quick Schedule Definition 367	Display Filters window
DESC	Fields 191
DO REMEDY parameter 460	Display Type
DESC Command	Active Environment Screen 169

Display Type A	DO Statement Action 434
Active Environment Screen 172	DO REMEDY
Display Type D	sending e-mail to Remedy Helpdesk 461
Active Environment Screen 170	DO REMEDY Statement
Display Type Field	DO Statement Action 434
Active Environment Screen 171	DO RERUN
Display Type Indicator	Description 462
Job Dependency Network 243	DO RERUN Statement
Displaying	CMEM On Spool Job 687
Job Statistics 115	DO Statement Action 434
Jobflow 115	RERUNMEM Parameter 593
Statistics 142	DO RULE Calls
DO Action	Nesting 708
DO Statement 136	DO RULE Statement
DO COND Parameter	AutoEdit 707
COND-NAME Subparameter 437	CONTROL-O Rule 681, 707
DATEREF Subparameter 437	Example 709
Long Condition Names 436, 439	DO SET
OPT Subparameter 438	Global Variables 271
DO COND Statement	DO SET Statement
CMEM Rule 681, 700	AutoEdit Statement 795
Conflicts 439	Description 465
Definition 436	DO Statement Action 434
DO Statement Action 434	Example 468
Example 441, 701	Global Variable 760
Logic 438	JCL Setup 765
	Local Variable 757
OUT Parameter 440, 566 Prerequisite Condition 70, 501, 701	SET VAR Parameter 467, 623
DO CTBRULE Statement	User-Defined Variables 747
Description 442 DO Statement Action 434	DO SHOUT Statement
	CMEM CONTROL-O 681
Example 443	CMEM Rule 710
DO FORCEJOB Statement	CONTROL-O 710
Active Jobs File 704	CTO282I Subparameter 711
CICS Job 706	DO Statement Action 434
CMEM On Spool Job 686	Example 474, 637, 714
CMEM Rule 681, 703	JES 711
Dataset Event 704	Route Codes 712
Description 444, 703	Shout Facility 469
DO Statement Action 434	DO Statement
Example 446, 705	CMEM Rule 681, 695, 699
Logic 704	CMEM Rule Definition 260
RERUNMEM Parameter 593	Description 434
DO IFRERUN Statement	Logic 435
Description 447	Post-Processing Parameters 136
DO Statement Action 434	Summary 387
Example 451	DO STOPCYCL
Job Rerun 226	Description 475
RERUNMEM Parameter 593	DO STOPCYCL Statement
Scheduling Definition 218	DO Statement Action 435
DO MAIL Statement	DO STOPJOB Statement
Description 452, 460	CMEM Rule 681, 715
DO Statement Action 434	Description 715
DO NOTOK Statement	Example 716
DO Statement Action 434	DO SYSOUT Statement
DO OK Statement	Archiving Facility 484
Description 457	Description 477

Diagram 481	SHOUT Parameter 628
DO Statement Action 435	DUEIN Field
Example 483	Job Dependency Network 244
Logic 480	DUEINCHK Parameter
Merging 481	CTMPARM 493
SYSOUT Parameter 483, 645, 650	DUEOUT Field
DOC Command	Job Dependency Network 244
Scheduling Definition 142, 145, 219	Dummy Class
DOC Lines	DO SYSOUT Statement 480
Scheduling Definition 145, 490	SYSOUT Parameter 647
Status Zoom Screen 490	DUMMY Job
DOCLIB Field	Status 389, 390
	DUMMY Jobs
Scheduling Definition 146 DOCLIB Parameter	
	MEMBLIB Parameter 390
Description 488	DUMMY Library
Example 489	MEMLIB Parameter 523
DOCMEM Field	OVERLIB Parameter 571
Scheduling Definition 146	DUMP Command
DOCMEM Member	Active Environment Screen 179
DOCLIB Library 488	duplicate dataset prevention 585
DOCMEM Parameter	DUPLICATE Field
Description 490	Scheduling Confirmation 154
Example 491	Dynamic Destination Table
DOCU/TEXT 488, 490	DO SHOUT Statement 472, 633, 713
Browse Mode 145	Dynamic Group Insert Facility 69
Interface 324, 372, 855	Dynamic insert job into group Field
INVOKE JOBSCAN Parameters 855	Scheduling Confirmation 154
JCL Documentation 372	
Online Utilities Menu 323	_
Option 323	E
Option U1 324	_
Simulation/Tape Pull 333	E Option
Tape Pull List 855	Active Environment Screen 182
Utility 324	Display Filters Window 301
Documentation	Job List Screen 123
AUTO-SAVE Field 120	Manual Conditions 294
AUTOSAVE Field 146	ECJ Task Type
DESC Parameter 432	IOA Log Show Screen Window 304
Editing 145	Show Screen Filter 196
Saving 146	ECS Task Type
Scheduling Definitions 144	IOA Log Show Screen Window 304
Double Confirmation Window 155	Show Screen Filter 196
DOWN Command	EDIT Command
PF08/PF20 93, 95	CMEM Rule Definition 262
DSN Parameter	Job Scheduling Definition 913
ON DSNEVENT Statement 725	Scheduling Definition 142
DSNEVENT criteria	Edit Entry Panel
STEPRC field 689	IOA Editor 102
	Edit Environment 913
DSNEVENT Statement 259	CMEM Rules 263
DUE IN Time	Description 913
DUE OUT Parameter 493	Example 918
DUE OUT Parameter	
Description 492	
	Line Editing Commands 143
Example 493	EDIT Option 182
Job Flow 75	EDIT Option 182 Active Environment Screen 182
	EDIT Option 182

Editing	Job Ordering 813
CMEM Rule Definition 263	Utility CTMJBINT 370
CMEM Rules 251	END_NOK_ABND Status
Documentation 145	CTMAPI 890
Example 918	END_NOK_CC Status
Job JCL 125, 182	CTMAPI 890
Rule Definitions 927	END_NOK_DISA Status
Scheduling Definition 143, 913	CTMAPI 890
EDMEM command	END_NOK_JCLE Status
IOA Editor 102	CTMAPI 890
ELAPS Field	END_NOK_NSUB Status 890
Job Dependency Network 244	END_NOK_UNKW Status
ELAPSE Field	CTMAPI 890
Zoom Screen 216	END_OK Status
ELAPSE TIME	CTMAPI 890
Job Flow 76	END_OK_FOK Status
Elapse Time	CTMAPI 890
DUE OUT Parameter 493	ENDED NOT OK - ABENDED Status
ELAPSED Field	Active Environment Screen 185
Job Order Execution History 236	ENDED NOT OK - DUE TO CC Status
ELAPSED Run Time Field	Active Environment Screen 185
Statistics Screen 240	ENDED NOT OK - JCL ERROR Status
ELAPSED Time, Group	Active Environment Screen 186
Statistics Screen 240	ENDED NOT OK - RERUN WAS NEEDED Status
Emergency Execution	Active Environment Screen 186
DASUBMIT DD Statement 794	ENDED NOT OK - TERM ON NCT2 Status
Emergency Job	Active Environment Screen 186
MAXWAIT Parameter 520	ENDED NOT OK Status
TASKTYPE Parameter 654	Active Environment Screen 185
EMR Task Type	TERMINATE Option 247
IOA Log Show Screen Window 304	ENDED NOTOK Status
Show Screen Filter 196	DO STOPJOB Statement 715
End Code	ENDED NOTOOK Status
ON Statement 550	Active Environment Screen 189
END Command	ENDED OK FORCED OK Status
Calendar Definition Screen 321	Active Environment Screen 186
CMEM Rule Definition 264	ENDED OK Status
IOA Log Show Screen Window 306	Active Environment Screen 186, 189
PF03/PF15 93	Job Graph 205
Scheduling Definition 148	Show Screen Filter 195
Show Screen Filter 199	TERMINATE Option 247
END NOT OK Status	ENDED Status
END NOTOK Status 166	Show Screen Filter 195
END NOTOK Field	ENTER Key
Global View Screen 200, 202, 204	AutoRefresh Mode 100
END NOTOK Status	Enter YES Field
Job Graph 203, 205	Simulation/Tape Pull 337
END OK Field	ENTER YES TO CONTINUE Parameter
Global View Screen 200, 202, 204	Description 328
Job Graph 203	Prerequisite Condition Utility 326
END OK Status	Entry Panel
ENDED OK Status 141	AutoSave Documentation 146
END TIME Field	Calendar Facility 308
Statistics Screen 240	CMEM Rule Definition 253
End TRACE level	CMEM Rule Facility 252
SET Command Panel 106, 107	Exiting 150
End User Job Order Interface	IOA 84

Parameter Prompting 337	Statistics Screen 238
Scheduling Facility 113, 116, 150	Execution Time
Table Creation 112	DUE OUT Parameter 493
Environment	EXERR Code
Online Facility 81	ON Statement Codes 552
<b>Environment Specification</b>	EXERR Status
SET VAR Parameter 624	Description 386
EQ Operator	ON Statement 548
AutoEdit Facility 773	Exit
ERASE Option	CTMX001 814
Manual Conditions 294	CTMX002 525, 599, 793, 837, 847
Errors Only Field	CTMX003 847
Simulation/Tape Pull 337	CTMX004 599
EST Task Type	CTMX013 240
IOA Log Show Screen Window 304	CTMX013 240 CTMX014 525
Show Screen Filter 196	Exit Command
Event Selection Parameter	
	Online Facility 91
CMEM Rule 259, 694	EXIT Option
Exclusive Control	Online Utilities Menu 323
CONTROL Parameter 408	Parameter Prompting Entry Panel 337
Exclusive Resource	Primary Option Menu 87
WAIT SCHEDULE Status 286	Exit Window
EXEC A PLAN Option	Job List Screen 149
Parameter Prompting 831	Rule List Screen 264
EXEC Phase	Exiting
Parameter Prompting 354	CMEM Entry Panel 266
EXEC Status	CMEM Rule Facility 264
CTMAPI 890	CMEM Rule List 264
Exec/Order a Plan	CMEM Table List 266
Parameter Prompting 356, 359	Define Parameters in Master Plan 355
EXEC_ERR Status	IOA Log Show Screen Window 305
CTMAPI 890	Job List 150
EXEC_INQ Status	Job Scheduling 148
CTMAPI 890	Quick Schedule Definition 368
EXEC_NJE Status	Scheduling Definition 148
CTMAPI 890	Show Screen Filter 198
EXEC_WSUB Status	Exits
CTMAPI 890	CTMX015C 572
EXECTIME Limit	CTMX015O 572
SHOUT Parameter 629	External Tape
Execute a Plan	Example 802
CTMEXEC CLIST 832	2
EXECUTING (SYSOUT IN HOLD STATUS)	
Active Environment Screen 186	F
EXECUTING Field	
Global View Screen 200, 202, 204	F Option
EXECUTING Status	Active Environment Screen 181
	Job List Screen 126
Job Graph 203, 205 Show Screen Filter 195	Table List Screen 116, 122, 151
	FAILED REASON UNKNOWN Status
Execution Delay	Activate Option 181
MAXWAIT Parameter 519	Fast Exit
Execution Error	Online Facility 91
ON Statement 550	Fetch a Plan
Execution Information	
Job Order Execution History 234	CTMFETCH CLIST 832
Statistics Facility 54	Parameter Prompting 355
Execution Statistics	FETCH A PLAN Option

Parameter Prompting 830	FORCED SCHEDULING Parameter 32
Field-Sensitive Help	Forcing Jobs
Online Facility 99	Overview 66
File Name	Forecasting
DO SYSOUT Statement 477	Overview 54
SYSOUT Parameter 644	Simulation 54
File Prefix	Form Type
Parameter Prompting 340	APPL FORM Parameter 395
File Transfer	FR Parameter
Example 702	STEP RANGE Parameter 641
File Transfer to PC	FRCOKOPT Parameter
PC PACKET STATUS Option 702	CTMPARM 247, 553
Filter	FREE Option
Job Dependency Network 242, 243	Active Environment Screen 181
FILTER Field	Free Tracks
Active Environment Screen 171	MINIMUM Parameter 531
IOA Log Show Screen Window 302	PDS Parameter 579
Show Screen Filter 193, 302	FRM Parameter
Filtering	DO SYSOUT Parameter 477
Active Environment Screen 189	FROM Class
Log Screen 299	DO SYSOUT Statement 477, 480
FIND Command	SYSOUT Parameter 644, 647
Description 96	FROM DATE Field
Graphic Jobflow Screen 162	Date Range Window 164
PF05/PF17 93, 163	FROM DAYS
FLUSH Code	TIME + DAYS Parameter 659
+EVERY Step 549	FROM Field
ON Parameter 553	Zoom Screen 216
FLUSH Value	FROM STEP Field
PREVENT-NCT2 Parameter 584	Restart Confirmation Window 226
FORCE Code	Restart Step List Window 234
ON Statement 548	From Step/Proc \$FIRST/\$CLEANUP
ONStatement Codes 552	Rerun/Restart Window 228
Force Job	FROM subparameter
CMEM 681	INTERVAL Parameter 514
FORCE OK	FROM TIME
ANYSTEP 247	TIME + DAYS Parameter 659
Force OK	FROM Time
%%JOBCC 754	TIME Parameter 659, 661
%%MAXRC 754	fromcol Parameter
Cyclic Jobs 247	FIND Command 97
Group Entity 248	FROMJOB Field
FORCE OK Confirmation Window	Quick Schedule Definition 364
Active Environment Screen 248	FTP products
Description 247	CMEM 691
FORCE OK Option	FUNCTION Field
Active Environment Screen 248	Prerequisite Condition Utility 325
FORCE Option	FUNCTION Parameter
CMEM Table List 255, 267, 682	AutoEdit Simulation 331
DO FORCEJOB Statement 444	Functions
Job List Screen 116, 126	IOA Primary Option Menu 86
Manual Scheduling 151	<i>y</i> 1
Table List Screen 116, 122, 151	_
FORCE# RT Installation Parameter 445, 705	G
FORCE#WI Installation Parameter 445, 705	•
FORCED FROM TIME Field	G Option
Parameter Prompting 357	Table List Screen 122
I O	

GDG Adjustment 585	Number of Characters 66
GE Operator	Gregorian Date Standards
AutoEdit Facility 773	Overview 65
General Job Parameter	GROUP Command
Scheduling Definition 130	Active Environment Screen 179
GENERAL Library	Log Screen 298
DALIB DD Statement 525	GROUP Criteria
MEMLIB Parameter 523	IOA Log Show Screen Window 304
OVERLIB Parameter 572	Group Entity
GENERAL Message Type	Deleting a Job 211
IOA Log Show Screen Window 303	Display 182
General Parameters	Group Scheduling Table 376
CMEM Rule 694	Group-Handled Jobs 68, 112
CMEM Rule Definition 260	ON GROUP-END Parameter 539
Summary 378	OUT Conditions 566
General Profile	Parameters 140
Active Environment Screen Filter 189, 299	Prerequisite Condition 71
Generation Data Sets	Scheduling Definition 138
CMEM 691	Scheduling Definition Screen 139
Generation Dataset (GDG) Adjustment 585	Statistics Screen 240
Generic Resource	Task Type 196
Example 804	Undeleting 182
GLOBAL Control Statement	Zoom Screen 218
%%GLOBAL 771	GROUP Field
Global Profile	Active Environment Screen 172
Customizing 81	Global View Screen 201
Global Variable	Group Entity 140
AutoEdit 746	Quick Schedule Definition 364
Backslash Character 754	Show Screen Filter 194
Distinguishing 761	Group Handled Jobs 112
JCL Setup 760	Group Name
Resolution 765	GROUP Parameter 495
Syntax 761	JOBSTART Command 142
Global Variable Database	GROUP NAME Field
Structure 760	View Graph Screen 203
Global Variable Extension 898	GROUP NAME Parameter
Global Variable Return Codes	AutoEdit Simulation 330
CTMAPI 899	GROUP Option
Global Variables	Active Environment Screen 182
Variable Database Facility 271	GROUP Parameter
Global View Screen	CMEM Rule 717
#END Field 201	Description 495
#EXC Field 201	Emergency Jobs 656
Active Environment Screen 180	Example 497
Fields 200	Group Entity 140
Group Statistics 199	MAXWAIT Parameter 520
GOING TO START Status	Group Scheduling
Active Environment Screen 186	Group Entity 69, 138
Graphic Display	Job List Screen 123
Job Status 180	Logic 382
GRAPHIC FLOW Option	MAXWAIT Parameter 520
Table List Screen 122, 162	Option O 151
Graphic Jobflow	Parameters 140
Display Width 162	Group Scheduling Table
Screen 162	Group Entity 376
Gregorian Date Format	Maybe Jobs 820
Definition 65	REMOVE UNSCHED CONDITIONS Field 820

Group Statistics	Description 49
Global View Screen 199	History Jobs file 400
View Graph Screen 201	History Jobs file and 400
Group Status	HLDCLAS Installation Parameter
Global View Screen 201	DO SYSOUT Statement 478
Group-Handled Jobs 112	SYSOUT Parameter 645
Description 68	HOLD Field
Groupname Argument	Scheduling Confirmation 154
JOBSTART Command 176	HOLD Option
GRP Entry	Active Environment Screen 180
Active Environment Screen 182	Host Node
GRP HELD Status	NJE Network 51
Active Environment Screen 186	TVIL TVCCWOLK 01
GRP MAXWAIT Parameter	
Description 498	
Group Entity 140	
GRP MAXWAIT parameter 499	I Option
GRP Task Type	Job List Screen 126
IOA Log Show Screen Window 304	Parameter Prompting 344
Show Screen Filter 196	I1 ISPF Utility
	Add Prerequisite Condition 324
GRPRECHK Parameter	Check Prerequisite Condition 324
CTMPARM 112	Delete Prerequisite Condition 324
GT Operator	Il Option
AutoEdit Facility 773	Online Utilities Menu 325
	IBM
III	3720 Terminals 82
H	IBM FTP
H Option	CMEM 692
Active Environment Screen 180	IDMS/DC 81
	PF06/PF18 93
HALF Page	
Scrolling Amount 95 HELD Class	IEF125I Message
	ON DSNEVENT Statement 726, 735
DO SYSOUT Statement 478 Held Class	IEF403I Message
	ON DSNEVENT Statement 726, 735
SYSOUT Parameter 645	IEFPROC Stepname ON DSNEVENT Statement 725
HELD Status	
Active Environment Screen 186	ON STEP Statement 734
CMEM On Spool Job 686	IF Logic
Job Deletion 211, 248	Example 808
Help	IGD101I message 691
Line Sensitive 99	IGD104I message 691
Online Help 100	IGD105I message 691
HELP Command	IGD107I message 691
PF01/PF13 93, 99	IGD108I message 691
Hexadecimal Value	IGD17101 message 691
Special Characters 82	IGNORE DSN Parameter
HIST Installation Parameter	Tape Pull List 854
CONTROL-M/Restart 49	IGNORE IN Parameter
HISTORY Command	IOALDNRS Utility 828
Active Environment Screen 175	IGNORE JOB Parameter
History Environment Screen 245	Tape Pull List 854
Options 246	Implementation
RESTORE Option 247	Job Scheduling 812
History Jobs File	Manual Conditions 817
# OF DAYS TO KEEP 603	Parameter Prompting 823
# OF GENERATIONS TO KEEP 605	IMS/DC 81

PF06/PF18 93	FROM Subparameter 514
IMSACTIVE	INTERVAL Subparameter 513, 51
Prerequisite Condition 569	RERUNMEM Parameter 594
IN Condition	Simulation 843
Erased Automatically 390	Simulation/Tape Pull 335
IOALDNRS Utility 817	TASKTYPE Parameter 655
Job Dependency 242	INTERVAL Subparameter
IN Field	INTERVAL Parameter 513, 514
Zoom Screen 216	INTRDR Internal Reader
IN Parameter	Submit Authority 333
COND-NAME Subparameter 500	Tape Pull List 852
DATEREF Subparameter 501	INVOKE JOB/SCAN
Description 500	Simulation/Tape Pull 336
Example 506	Tape Pull List 855
Logic 503, 509	IOA
Long Condition Names 500, 502	Conditions File 62
Quick Schedule Definition 361	Customization 80
IN PROCESS Status	Display Format Members 81
Show Screen Filter 195	Log File 62
IN Statement	Manual Conditions File 62
Manual Conditions 291	Primary Option Menu 85
Prerequisite Condition 70	Under ISPF 101
INCLIB Control Statement	IOA Calendar Facility
%%INCLIB 775	Calendar Facility 306
INCONTROL	IOA Conditions
Core Description 62	File 281
INFO Command	IOA Conditions File 282
Primary Option Menu 87	IOA Conditions/Resources
Information about Job	Screen 281
DESC Parameter 432	IOA Conditions/Resources File 282
Input Files	IOA Conditions/Resources Screen
Simulation 841	Description 281
Input Registers	Manually Releasing Jobs 390
CTMAPI 883	IOA Core
INQ/UPD MEDIA DB Option	Description 62
Primary Option Menu 89	IOA Editor 102
INSERT BY WEEK DAYS Option	Command line commands 104
Year List Screen 313	Edit Entry Panel 102
Insert Command	EDMEM command 102
Edit Environment 916, 929	PFKey functions 103
INSERT Option	Row commands 104
CMEM Rule List 257	IOA Editor screen 103
Job List Screen 113, 126	IOA Entry Panel 84
Parameter Prompting 344, 355	IOA Global Variable Database
Year List Screen 312	AutoEdit 746
Inserting	Structure 760
Relevant Screen or specific item to insert 313	IOA Log Facility 296
Inserting Additional Job 69	Description 50
Installation Working Date	Post-processing 50
Working Date 750	IOA Log Screen
INSTREAM JCL Parameter	DISPLAY Command 298
Description 511	Messages 305
Interval	IOA Log Show Screen Window
Periodic Calendar 317	Activating 300, 301
INTERVAL Parameter	Activating Filters 300
Description 513	Active Environment Screen 301
Example 515	DESC 302
Zamipio viv	D100 00W

Exiting 305	Scheduling Definition 110
Fields 302	ISPF SPLIT Command
Message Types 305	PF02/PF14 93
IOA Manual Conditions 60	ISPF/PDF Facilities
IOA Manual Conditions Screen	Online Facility 108
Manually Releasing Jobs 390	ISPF/PDF Primary Option Menu
IOA Primary Option Menu 85	ISPF KEYS Command 101
Option 6 323	ISPSTART Command
Option 7 291	ISPF Keys 101
Option F 89	IV Option
Option S 296	IOA Primary Option Menu 272
PC PACKET STATUS 89	
IOA SAMPEXIT Library	
SAMPEXIT Library 525	J
IOA SET 105	
IOA Variable Database 898	J Option
IOA Variable Database Facility 271	Active Environment Screen 182
Entry Panel 272	Job List Screen 126
IOA Variables Database	JCL
CTMAPI and 865	Editing 182
IOA Variables Facility	in job definition 511
Entry Panel 272	instream 511
IOA125I message 689, 690	JCL Check Field
IOA283I message 690	Simulation/Tape Pull 337
<u> </u>	JCL Command
IOA285I message 690	Job List Screen 115
IOA287I message 690	JCL Documentation
IOA403I message 689, 690	DOCU/TEXT Product 372
IOAAPI Functions 865	
IOADFLT Parameter	JCL Edit
IOAENV Library 423	Active Environment Screen 182
IOADLD Utility	Job List Screen 126
Variable Database Facility 278	JCL Error
IOADUL Utility	Intentional 766
Variable Database Facility 278	ON Statement 550
IOAID Field	JCL ERROR Status
Conditions/Resources 283	Show Screen Filter 195
IOALDNRS Utility 291	JCL Expanded
Manual Conditions 817, 825	SYSDATA 68
Parameter Prompting 828	JCL Library
IOALog Screen 60	CTMQSB Command 814
IOANOTE Utility	OVERLIB Parameter 571
Tasktype WRN 656	PPF2DAY Job 838
IOASE07 Security Exit 900	PPF2DEL Job 838
IOAUTIL CLIST	JCL Library Mode
	AutoEdit Syntax Testing 329, 793
Online Utilities 323	Parameters 330, 729
IOAVAR	JCL LIBRARY Parameter
Variable Database Facility 271	AutoEdit Simulation 330
IOAVARLD Job	
Variable Database Facility 278	JCL Management
IOAVARUL Job	CMEM On Spool Job 687
Variable Database Facility 278	JCL Member
ISPF 95	OVERLIB Library 571
AutoRefresh Mode 101	OVERLIB Parameter 331
PACK Option 307	RERUNMEM Parameter 593
ISPF Commands	JCL Modification
Priority 101	OVERLIB Parameter 571
ISPF PACK Option 528	JCL Option
•	

Active Environment Screen 182	DO SYSOUT Statement 478
Job List Screen 126	SHOUT Parameter 630
JCL SET	SYSOUT Parameter 645
AutoEdit SET Statement 770	JES3
JCL Setup	cpuid 524
%%ELSE Control Statement 772	DO SHOUT Statement 470, 711
%%ENDIF Control Statement 772	DO SYSOUT Statement 478
%%GLOBAL Control Statement 771	SHOUT Parameter 630
%%GOTO Control Statement 772	SYSOUT Parameter 645
%%IF Control Statement 772	JESDS Subparameter
%%INCLIB Control Statement 775	SYSDATA Output Class 68
%%INCMEM Control Statement 775	JESYSMSG 688, 690
%%LABEL Control Statement 772	JFAIL Code
%%LIBSYM Control Statement 776	ON Statement Codes 552
%%MEMSYM Control Statement 776	JFAIL Status
%%RANGE Control Statement 777	Description 386
%%RESOLVE Control Statement 778	JLOST Code
%%SET Control Statement 779	ON Statement Codes 552
AutoEdit 46, 781	JLOST Status
Control Statement 770	ON Statement 548
CTMAESIM Utility 793	JNRUN Status
Date Variable 750, 797	Description 386
DO SET Statement 766	ON Statement 548
	JNSUB Code
Global Variable 760	ON Statement Codes 552
Local Variable 757	
Modification 571, 745	JNSUB Reason
Nested Expressions 774	NOT SUBMITTED Status 766
Operators 773	JNSUB Status
Syntax Checking 793, 851	ONStatement 548
Sysout Archiving 796	Job
System Variable 748	Displaying Jobflow 115
User-Defined Variable 754	Displaying Statistics 115
Variable Resolution 767	Job Activation Option
Work Flow 765	Active Environment Screen 181
JCL Statement	Job Arrival Event
MEMNAME Parameter 528	CMEM 680, 729
Syntax Checking 793, 851	DO FORCEJOB Statement 704
JCL statement	Job Arrival Rules
limitations 511	CMEM On Spool Job 687
JCL Syntax	Job Chain
Checking 793, 851	DO COND Statement 441
CTMSCIM Utility 333	IN Parameter 508
JCLFILE Parameter	Job Copying 156
DAJCLOUT DD Statement 855	Job Deletion
Tape Pull List 854	Active Environment Screen 211, 248
JES HOLD Command	Undeleting 182
Job Status 186	Job Dependency
JES Initialization	%%PLANID 834
SYSDATA Output Class 68	Critical Path 74
JES Instruction	Job Flow 75
DO SYSOUT Statement 480	Maybe Job 819
SYSOUT Parameter 647	Predecessor/Successor Job 75
JES Spool	Prerequisite Condition 71, 501
ON JOBARRIV Statement 729	REFRESH Command 176
JES2	Job Dependency Network
cpuid 524	Commands 244
DO SHOUT Statement 470, 711	Description 242

NET Option 181	On Spool Job 684
Quick Schedule Definition 365	Job Order Execution History Screen
Job Documentation	Active Environment Screen View Option 181
DESC Parameter 432	Description 234
Documentation 144, 145	Fields 235
Job End Event	JOB ORDER ISSUE Option
CMEM 680, 721	Online Utilities Menu 323
Job Execution Time	
DUE OUT Parameter 493	Job Ordering 150 Definition 66
Job Filter	
	End User Job Order 370
Log Screen 213	Example 801 Job Order Panel 813
Job Flow	
Adjustment 58, 74	Order ID 68
ELAPSE Time 76	Quick Submit Command 813
Manual Modification 76	Special Purpose Job 815
Job Flow Report	Utility 326
Prerequisite Condition 75	JOB Parameter
Job Forcing	DO FORCEJOB Statement 444, 703
CMEM On Spool Job 687	ON STEP Statement 733
Definition 66	Job Parameters
Logic 687	Scheduling Definition 130
Job Graph	Job Priority
ENDED OK Status 247	Priority 74
JOB GRAPH Line	Job Production
View Graph Screen 203, 204	Scheduling 377
Job Interdependency	Job Reactivate Option
Job Dependency 365	Active Environment Screen 181
Job List Exit Window	Job Request Utility Screen Parameters 327
Table Creation 113	Job Rerun 67
Job List Screen	MAXRERUN Parameter 516
Commands 124	Job Restart 67
COPY Option 115	Job Run Statistics
Copying Jobs 156	Statistics Screen 238
Delete Job 123	JOB SCAN
Description 123	AutoEdit Simulation 332
Edit JCL 126	JOB SCHEDULE DEF Option
Exiting 149	IOA Primary Option Menu 87
Fields 366	Job Scheduling 150
Format 124	Alternative Methods 812
Job Ordering 813	AutoEdit Facility 747, 795
Manual Job Scheduling 150	CTMAJO Program 815
Options 115, 125, 367	CTMJOB 812
Quick Schedule Definition 365	CTMQSB Command 813
Scheduling Definition 113	Implementation 813
Job Log	Screen 127
SYSDATA 68	Special Purpose Job 812
Job Name	Table List Screen 813
Example 804	Job Scheduling Definition
Job Scheduling Plan Screen 166	Calendar Facility 52
MEMNAME Matching 687	CMEM Rule 685
JOB NAME Parameter	Commands 142
AutoEdit Simulation 331	Group-Handled Jobs 112
Description 327	New Day Processing 46
ON JOBARRIV Statement 729	Parameters 44
ON JOBEND Statement 731	Storing 44
Parameter Prompting 353	job scheduling definition
Job On Spool	DOC Lines 490

Job Scheduling Plan	Jobs Left Field
Calendar Format 164	Simulation/Tape Pull 336
Screen 165, 166	JOBSTAT Command
Job Status	Active Environment Screen 176
Active Environment Screen 185	Job Scheduling Definition Screen 142
Description 385	JODID Field
JOB STATUS Field	Job Order Execution History 236
Global View Screen 201	Joining
JOB STATUS Option	Concatenation 768
IOA Primary Option Menu 87	Journal File
Job Submission	Overview 62
Manual Confirmation 406, 449	Journaling
Scheduling Criteria 47	Description 49
Job Sysout	JRNL Installation Parameter
Sysout 68	Journaling 50
JOB Task Type	JSECU
IOA Log Show Screen Window 304	ON Statement Codes 552
Show Screen Filter 196	JSECU Status
Job Task Type	ON Statement 548
TASKTYPE Parameter 654	JTYPE Parameter
Job Termination	ON DSNEVENT Statement 724
DO STOPJOB Statement 715	ON JOBARRIV Statement 729
ON JOBEND Statement 731	ON JOBEND Statement 731
JOB TYPE Parameter	ON STEP Statement 733
ON JOBARRIV Statement 729	Julian Date
Job Type Parameter	JCL Setup 750
ON JOBEND Statement 731	Julian Date Format
Job Undeleting 182	Definition 66
JOB WAIT FOR PIPES COLLECTION	Julian Date Standards
Why Screen 206 JOB/SCAN	Overview 66
INVOKE JOBSCAN Parameters 855	1/
Simulation/Tape Pull 333	K
Tape Pull List 855	W l lCl .
JOB/SCAN Product	Keyboard Character
AutoEdit Simulation 332	Hexadecimal Value 82
JOBARRIV Statement 259	KEYS Command
JOBEND Statement 259	ISPF/PDF Primary Option Menu 101
Jobflow	KILL option 185
Graphic Display 162	KOA Recorder Option
JOBID Field	Primary Option Menu 89
Active Environment Screen 171	KSL ADDMNCND Utility
Statistics Screen 239	Maybe Jobs 820
Zoom Screen 215	J.
JOBID, Group	
Statistics Screen 240	
JOBNAME Criteria	E
· · · - · · - · · - · · · · · · · ·	L Command
IOA Log Show Screen Window 304	Parameter Prompting 343, 346, 354
JOBNAME Field	L Option
Active Environment Screen 171	Active Environment Screen 180, 212
Job Order Execution History 236	LABEL Control Statement
Statistics Screen 239	
Zoom Screen 215	%%LABEL 809
JOBNAME Parameter	Last Working Date
ON DSNEVENT Statement 724	Example 800, 806
JOBNAME, Group	LATE EXECUTION Status
Statistics Screen 240	Active Environment Screen 186

LATE Field	Log File 62
Job Dependency Network 244	CONTROL-M Log Screen 212
Show Screen Filter 192	IOA Log Screen 296
LATE Status	LOG Mode
Active Environment Screen 186	CMEM Rule Simulation 719
LATE SUBMISSION Status	LOG Option
Active Environment Screen 186	Active Environment Screen 180, 212
LATESUB Value	Primary Option Menu 86
SHOUT Parameter 628	Log Screen
LE Operator	<b>CATEGORY Command 298</b>
AutoEdit Facility 773	Commands 297
Leap Year	CONTROL-M Log Screen 212
Definition 784	Description 296
LEFT Command	Example 92
PF10/PF22 93	Fields 297
LEVEL Field	Filtering 299
Job Dependency Network 244	GROUP Command 298
LIBRARIAN 525, 572	IOA Log Screen 296
Job Documentation 146	Job Messages 296
Library	MESSAGE TYPE Codes 303
Maintenance 531, 579	Overview 60
LIBRARY Field	Stacking Multiple Jobs 213
Calendar Facility Entry Panel 309	Long Condition Names
CMEM Entry Panel 253	DO COND Parameter 436, 439
CMEM Rule Exit Option 265	IN Parameter 500, 502
Exit Option Window 149	OUT Parameter 562, 565
Parameter Prompting 341, 351	Long Prerequisite Condition
Quick Schedule Definition 362	Adding 285
LIBRARY Parameter	LPAR Field
DO FORCEJOB Statement 444, 703	Zoom Screen 217
DO RULE Statement 707	LT Operator
Save Documentation Window 147	AutoEdit Facility 773
Line Editing	
Edit Environment 913, 915, 927, 928	R.A.
Example 918	M
Job List Screen 367	M IOR Mossago Typo
Line Editing Commands 915, 928	M JOB Message Type IOA Log Show Screen Window 303
Line Number Field	M-INT Message Type
Quick Schedule Definition 366	IOA Log Show Screen Window 303
Linking Concatenation 767	M1 Option
LIST Function	Online Utilities Menu 326
AutoEdit Simulation 331	M2 Option
LIST Value	Online Utilities Menu 328
PREVENT-NCT2 Parameter 584	M3 Option
LOADGLOBAL Operator Command	Online Utilities Menu 332, 854
Variable Database Facility 281	M4 Option
Loading to Cache	Online Utilities Menu 337
%%GLOBAL Members 758	M5 Option
Local Variable	Online Utilities Menu 360
AutoEdit 746	M6 Option
Distinguishing 761	Online Utilities Menu 370
JCL Setup 757	Mail Prefix Value
LOCATE Command	DO SHOUT Destination 471, 711
Description 96	MAILDEST 472, 633
Location Commands	MAILDEST table 472, 633
Edit Environment 916, 930	Main Menu

IOA Primary Option Menu 85	MAX RC Field
Maintenance	Job Order Execution History 236
Libraries 531, 579	MAXCCOK Parameter
MAINVIEW Batch Optimizer (MVBO)	CTMPARM 552, 553
CONTROL-M Support 56	MAXDAYS Parameter
Implementation 821	AUTOARCHIVE Parameter 399
PIPE Parameter 581	MAXRERUN Limit
System-Related Considerations 823	Manual Job Rerun 222
MAINVIEW Batch Optimizer (MVBO) Option	MAXRERUN Parameter
Active Environment Screen 185	Description 516
Manual Conditions	RERUNMEM Parameter 594
Add Condition 293	MAXRUNS Parameter
Description 291	AUTOARCHIVE Parameter 399
Fields 292	MAXWAIT Parameter 521
Loading 817	Basic Scheduling Criteria 383
Maybe Job 817	Description 519
Options 294	Example 521
Overview 60	Maybe Dependency
Unscheduled Condition 817	Maybe Job 73
Manual Conditions File 62	Unscheduled Condition 819
Manual Intervention 817	Maybe Job
Unscheduled Conditions 818	@ OUT Conditions 819
Manual Confirmation	ADDMNCND Utility 820
CONFIRM Parameter 406	
DO IFRERUN Statement 449	Group Scheduling Table 820
Rerun Confirmation 222	Job Dependency 73 Manual Conditions 817
Restart Confirmation 224	Prerequisite Condition Prefix 819
Manual Intervention	MCT
IN Parameter 502	Simulation 847
OUT Parameter 564	MCTSMIND
Prerequisite Condition 61, 72	Simulation 847
Manual Job Ordering 150	MEM/MIS Criteria
Manual Job Release 390	IOA Log Show Screen Window 304
Manual Job Scheduling 150	MEMBER Field
Manual Rerun	WHY Option 290
MAXRERUN Parameter 517	MEMBER NAME Parameter
Manual Reruns	AutoEdit Simulation 330
Rerun Confirmation 222	MEMBER Parameter
Restart Confirmation 224	Save Documentation Window 147
Masking	Member Specification
Description 83	%%GLOBAL Control Statement 771
ON Statement 723	MEMBLIB Parameter
ON Statement CODES 555	PSEUDO Jobs 390
Master Console	MEMLIB Library
SHOUT Parameter 629	COPMEM2O Parameter 572
Master Plan	JCL Member 182
Parameter Prompting 350, 829	MEMLIB Parameter
Master Plan PREFIX	AutoEdit Variable 796
Parameter Prompting 356	Daily JCL Library 836
Master Scheduling Table	Description 523
Parameter Prompting 829	DUMMY Jobs 390
Master Table Creation	Example 526
Parameter Prompting 340	Job 523
MAX	OVERLIB Parameter 571
Scrolling Amount 96	Started Task 524, 526
MAX Field	System Variables 747
Conditions/Resources 283	MEMNAME Criteria

Active Environment Screen Filter 193	MONTHS Parameter 533
IOA Log Show Screen Window 304	PDS Parameter 579
MEMNAME Field	WDAYS Parameter 673
CMEM Rule 685	MINUS Operator
Global View Active Environment Screen 201	%%MINUS 781
Group Entity 140	Minus-One
Job Name Matching 687	Multiple CONTROL-M Monitors 55
Job Order Execution History 235	Minus-One Support
Quick Schedule Definition 366	CTMPLEX 55
Show Screen Filter 193	Sysplex 55
MEMNAME Parameter	MISSION DEF Option
D-CAT Parameter 416	Primary Option Menu 88
Description 528	MISSION STATUS Option
Example 529	Primary Option Menu 88
Group Entity 140	Mission, CONTROLM/Analyzer
MAXRERUN Parameter 517	CTB STEP Parameter 413
OVERLIB Library 571	MODE Parameter
Scheduling Definition 126	CMEM Rule 719
MEMNAME Value	Example 720
CMEM On Spool Job 685	Monochrome Terminal
DOCMEM Default 490	Color Support 82
Job Scheduling 816	Graphic Jobflow Screen 163
Simulation 844	Month Field
MEMSYM Control Statement	Job Scheduling Plan Screen 166
%%MEMSYM 776	MONTHS Parameter
Message Content	DATES Parameter 418, 533
9	
Group Name 496 MESSAGE Field	Description 533 MINIMUM Parameter 531
	PDS Parameter 579
Log Screen 297	Periodic Value 424
Parameter Prompting 354, 359	_
Message File	Move Commands
Simulation 846	Edit Environment 916, 929
Message Generation	MPP  Master Promoting Plan 990
DO SHOUT Statement 469	Master Prompting Plan 829
DO Statement 434	MS Parameter SHOUT Parameter 631
SHOUT Parameter 627	
Message Handling	MSG Library
Log File 60, 296	Help Member 100
Shout Facility 48	MSG STATISTICS Option
Message Line	IOA Menu 89
Online Facility 92	MSGCLASS Parameter
MESSAGE Parameter	SYSDATA Output Class 68
DO SHOUT Statement 712	MSGCLASS Sysout
Message Type	CMEM On Spool Job 684
IOA Log Show Screen Window 305	DO SYSOUT Statement 480
Message Type Criteria	SYSOUT Parameter 644, 647
IOA Log Show Screen Window 303	MSGLEVEL=1,1
MESSAGE TYPE Field	ON DSNEVENT Statement 726, 735
Screen Filter 303	MSTJCLB Parameter
Messages	CTMFETCH CLIST 833
Log File 305	Multi-Screen Control
MINIMUM Parameter	Transfer Command 90
CONFCAL Parameter 404	MVS MODIFY Command
DATES Parameter 418	protecting 269
DAYS Parameter 424	MVS MODIFY command
Description 531	Order or Force request 269
Example 532	

N	NJE Network
	CONTROL-M Monitor 51
N Option	NJE NODE Parameter
Active Environment Screen 181	Format 535
N Qualifier	Under JES2 535
DO Statement 555	Under JES3 536
NAME Field	node ID
Active Environment Screen 171	JES2 524
AutoEdit variable in 294	NODE NAME Field
Change Resource Window 289	Zoom Screen 217
Job Dependency Network 244	Non-Color Display
Manual Conditions Window 294	Monochrome Terminal 82
NAME Parameter	Non-periodic Calendar
CTB STEP Parameter 413	DCAL Parameter 422
NE Operator	WCAL Parameter 670
AutoEdit Facility 773	Non-periodic Scheduling
Nested Expressions	Format 670
JCL Setup 774	NOT CATLGD 2
NET Argument	CMEM 680
REFRESH Command 176, 245	DO STOPJOB Statement 716
	Job Status 186
NET Option	NOT CATLGD2 error prevention 585
Active Environment Screen 181, 242 NEW COND Command	NOT FOUND Status
Manual Conditions 293	Active Environment Screen 187
	NOT OK Status
New Day Procedure	Show Screen Filter 195
"Shifted" for SHOUT purposes 632	NOT STARTED Status
ODATE 64	Active Environment Screen 187
SHOUT jobs 632	NOT SUBMITTED Status
SHOUT WHEN LATE Message 632	Active Environment Screen 187
SHOUT WHEN LATESUB Message 632	JNSUB Reason 766
New Day Processing	
Description 46	NOT_DELETED Status
NEW LCOND Command	CTMAPI 890
Manual Conditions 293	NOTE
NEW PASSWORD Field	Zoom Screen 216
IOA Entry Panel 85	NOTE Command
NEWJOB Parameter	Active Environment Screen 177
Simulation 844	Zoom Screen 220
Next	NOTE Field
SAC Parameter 610	Active Environment Screen 172
NEXT Command	NOTE Status
Job Scheduling Plan 165	Active Environment Screen 187
PF11/PF23 93	NOTOK
Scheduling Definition 143, 149	Description 455
NEXT Value	NOTOK Status
Schedule Date 437, 563	Description 386
NEXTYEAR (PF11/PF23) Command	Group Entity 139, 141
Calendar Definition Screen 321	ON Statement 548
Night Schedule Field	NOTOK Value
Simulation/Tape Pull 336	ON Statement Codes 552
NJE Enhanced Tracking Support 688	SHOUT Parameter 628
NJE Field	NR Field
Zoom Screen 217	Quick Schedule Definition 366
NJE Job	NXT RUN Field
CMEM On Spool Job 683	Zoom Screen 217
NJE JOB Status	

**Active Environment Screen 187** 

0	ON CODE Parameter
<b>U</b>	ON Statement 537, 542
O Option	ON DSNEVENT Statement
Active Environment Screen 247	And/Or/Not Parameter 726
Job List Screen 126	CMEM Parameters 694
Table List Screen 151	CMEM Rule 680, 682, 721, 724
OBJECTS Option	CMEM Rule Definition 259
IOA Primary Option Menu 89	CMEM support 688
OCCUR NO Suffix	Dataset Event 724
AutoEdit Parameter 837	Example 728
OCCUR NO. Field	MSGLEVEL=1,1 726, 735
Parameter Prompting 353, 359	RUNTSEC=TRIGGER 739
ODAT	ON GROUP-END Parameter
IN Parameter 501	Group Entity 377
	ON GROUPEND Parameter
Prerequisite Condition 71, 72	Definition 539
Schedule Date 437, 563	Group Entity 141
ODATE	ON JOBARRIV Rule
Assignment 64	CMEM Rule 683
DAYTIMEM Parameter 64	DO FORCEJOB Statement 705
Definition 63	ON JOBARRIV Statement
DO FORCEJOB Statement 444	CMEM Parameters 694
Example 797	
GRP MAXWAIT Parameter 498	CMEM Rule 680, 684, 721, 729
Job Eligibility 64	CMEM Rule Definition 259
MAXWAIT Parameter 519	Example 730
Meaning 64	Job Arrival 680
New Day Procedure 64	ON JOBEND Statement
RUN Attribute 65	CMEM Parameters 694
System Variable 750	CMEM Rule 680, 721, 731
VALUE Attribute 65	CMEM Rule Definition 259
ODATE Field	Example 732
Active Environment Screen 171	Job End 680
Global View Screen 201	RUNTSEC=TRIGGER 739
Job Order Execution History 235	ON OUTPUT Q Status
Log Screen 297	Show Screen Filter 195
Parameter Prompting 357	ON OUTPUT QUEUE Status
Scheduling Confirmation 153	Active Environment Screen 187
Show Screen Filter 197	ON PGMST ANYSTEP
Zoom Screen 215	DO CTBRULE Statement 442
ODATE Parameter	ON Statement 542
AutoEdit Simulation 331	ON PGMST Parameter
OF Field	ON Statement 537, 542
Parameter Prompting 347	ON PGMST Statement
OIDL Command	Combinatorial Logic 218
Active Environment Screen 179	Step Range 641
OK Status	ON PGMST trigger
Description 386	Zoom Screen 218
Group Entity 139, 141	On Spool Job
ON Statement 548	CMEM Facility 52
Post-Processing Parameters 386	CMEM Rule 681
OK Value	Components 684
ON Statement Codes 552, 553	DO FORCEJOB Statement 705
SHOUT Parameter 628	Forcing Logic 686
OLDJOB Parameter	Job Flow 685
Simulation 844	NJE Job 683
ON Block	ON JOBARRIV Statement 730
	Status 188
ON Statement 545	Status 100

TYPERUN=HOLD 684, 688	OPT Subparameter
ON Statement	DO COND Parameter 438
* Character 547	OUT Parameter 563
+EVERY 544	Option?
CMEM Parameters 694	Active Environment Screen 180, 205
CMEM Rule 721	Option 1
CMEM Rule Definition 259	Parameter Prompting Entry Panel 337
Codes 550	Parameter Prompting Type 1 Menu 339
CODES Parameter 544	Parameter Prompting Type 2 Menu 349
Combinatorial Logic 722	Option 2
Conditional Processing 386	IOA Primary Option Menu 87
	v -
Description 537, 542	Parameter Prompting Entry Panel 337, 349
Example 547, 556	Parameter Prompting Type 1 Menu 339, 349
Logic 544, 555	Parameter Prompting Type 2 Menu 349, 355
Masking 723	Primary Option Menu 58, 111, 116
Multiple 545	Option 3
PGMST Parameter 543	IOA Primary Option Menu 87
PROCST Parameter 544	Parameter Prompting Type 2 Menu 349, 356
Specified Step 547	Primary Option Menu 58, 166
ON Statement Codes	Option 4
***** Code 551	Primary Option Menu 86
FORCE Code 552	Option 5
ON STEP Statement	Primary Option Menu 60, 86, 296
CMEM Parameters 694	Option 6
CMEM Rule 680, 733	Online Utilities 102
CMEM Rule Definition 259	Primary Option Menu 61, 86, 323
CMEM support 688	Option 7
ON SYSOUT Parameter	Primary Option Menu 60, 86, 291
Description 559	Option 8
Online Facility	Primary Option Menu 61, 86
Active Environment Screen 166	Option A
Documentation 144	Active Environment Screen 181
Exiting 91	Job List Screen 367
Help Screen 99	Manual Conditions 294, 818
Overview 57, 80 Tracking and Control 58	Parameter Prompting 344, 355
Tracking and Control 58	Primary Option Menu 88
TSO Application 109	Why Screen 208
Under ISPF 101	Option B
Online Utilities Menu	CMEM Table List 255
Utility Screen 61, 323	Job List Screen 368
OP Parameter	Table List Screen 121
SYSOUT Parameter 644	Option BA
OPER Value	Primary Option Menu 89
DO SHOUT Destination 470	Option BB
SHOUT Parameter 629	Primary Option Menu 89
OPER2 Value	Option BM
DO SHOUT Destination 470	Primary Option Menu 89
OPT Command	Option BR
Active Environment Screen 174	Primary Option Menu 89
OPT Field	Option BV
Conditions/Resources 283	Primary Option Menu 89
Manual Conditions 292	Option C
Rule List Screen 255	Active Environment Screen 182
OPT Parameter	IOA Primary Option Menu 87
DO SYSOUT Statement 477	Job List Screen 126, 257, 368
opt Parameter	Primary Option Menu 60, 250, 252
DO COND Statement 438	Option Code

DO SYSOUT Statement 477	Online Utilities Menu 323, 337
SYSOUT Parameter 644	Option M5
Option D	Online Utilities Menu 323, 360
Active Environment Screen 181, 247	Option M6
CMEM Rule List 257	Online Utilities Menu 323, 370
CMEM Table List 255, 266	Option N
Conditions/Resources 287	Active Environment Screen 181, 242
Job List Screen 125, 367	Option O
Parameter Prompting 344, 354	Active Environment Screen 181
Table List Screen 122, 158	Group Scheduling 151
Why Screen 208	Job List Screen 126
Option E	Manual Scheduling 151
Active Environment Screen 182	Restart Step List Window 234
Display Filters window 191	Option OA
Manual Conditions 294	IOA Menu 89
Option F	Option OC
Active Environment Screen 181	Primary Option Menu 89
AutoEdit Variable 796	Option OK
CMEM Table List 255, 267	Primary Option Menu 89
IOA Primary Option Menu 89	Option OL
Job List Screen 126	IOA Menu 89
Manual Scheduling 151	Option OM
Primary Option Menu 88	IOA Menu 89
Restart Step List Window 234	Option OR
Table List Screen 115, 122	IOA Menu 89
Option Field	Option OS
Active Environment Screen 171	IOA Menu 89
Job Dependency Network 243	Option P
Quick Schedule Definition 366	Job List Screen 126, 164, 367
Option G	Option R
Active Environment Screen 182	Active Environment Screen 181, 222, 225
Table List Screen 122, 162	Job List Screen 367
Option H	Parameter Prompting 344, 354
Active Environment Screen 180	Primary Option Menu 88
Option I	Option R1
CMEM Rule List 257	Online Utilities Menu 323
Job List Screen 126, 367	Option R2
Parameter Prompting 344, 355	Online Utilities Menu 323
Option I1	Option S
Online Utilities Menu 323, 325	Active Environment Screen 181
Option IV	CMEM Rule List 257
IOA Primary Option Menu 87	CMEM Table List 255
Option J	Display Filters window 191
Active Environment Screen 182	End User Job Order 370
Job List Screen 126	Job List Screen 125
Option L	Job Order Execution History 236
Active Environment Screen 180, 212	Table List Screen 121
Option M Job List Screen 368	Option T Job List Screen 126
Primary Option Menu 88 Option M1	Primary Option Menu 88
=	Restart Step List Window 234
Online Utilities Menu 323, 326	Option TC
Option M2	Primary Option Menu 89
Online Utilities Menu 323, 328	Option TI
Option M3	Primary Option Menu 89
Online Utilities Menu 323, 332, 854	Option TP  Primary Option Many 80
Option M4	Primary Option Menu 89

Option TR	JCL 126, 182
Primary Option Menu 89	Job Activation 181
Option TV	Job List 114, 125, 367
Primary Option Menu 89	JOB ORDER ISSUE 323
Option U	Job Reactivate 181
Active Environment Screen 182	JOB SCHED DEF 87
Primary Option Menu 88	JOB STATUS 87
Option U1	JOB/PIPE ACTIVITY 88
Online Utilities Menu 323, 372	KOA Recorder 89
Option V	LOG 180, 212
Active Environment Screen 181, 234	M4 on Online Utilities Menu 337
Option W	M6 on Online Utilities Menu 370
Active Environment Screen 185	MAINVIEW Batch Optimizer (MVBO) 185
Option X	MANUAL COND 86
Online Utilities Menu 323	Manual Conditions 294
Primary Option Menu 87	Master Plan 350
Option Z	Master Table Creation 340
Active Environment Screen 180	MSG STATISTICS 89
Options	NET 181
Activate 181	ORDER 116, 126, 151, 255
Active Environment Screen 180	PACK 110
AUTOMATION LOG 89	Parameter Definition 340
AUTOMATION LOG 89 AUTOMATION OPTION 89	
BALANCING DEF 89	Parameter Prompting 323, 340, 344
	Parameter Updating 340
BALANCING STATUS 89	PLAN 126, 164
CHANGE 287, 288	Prerequisite Condition 323, 340
CLEANUP 323	QUICK SCHEDULE 323
CMEM 250, 252	Reactivate 181
CMEM DEFINITION 87	RERUN 181, 222
CMEM Rule List 257	RULE ACTIVITY 89
CMEM Table List 254	RULE DEFINITION 89
CONFIRM 182	RULE STATUS 88, 89
COSMOS Status 89	Table List Screen 121
Cross Memory 93	TERMINATE 181
CTMEXEC 350, 356	UNDELETE 182
CTMFETCH 350, 355	UPDATE 344, 355
CTMQUICK 361	USER INTERFACE 323
DB VARIABLE DEF 89	VARIABLE DATABASE 87
Define Parameters and Conditions 341, 344	VIEW 181, 234
Define Parameters in Master Plan 355	WHY 180
DEL 181	Year List Screen 312
DELETE 115, 122, 125, 158, 255, 257, 266, 287, 344, 354	ZOOM 180
Delete Condition/Resource 287	Order a Job
Display Filters window 191	DO FORCEJOB Statement 444
DOCU/TEXT 323	DO Statement Action 434
EDIT 126, 182, 191	Next Day 801
EXIT 323	Order Daily Jobs Field
FORCE 116, 122, 126, 151, 267	Simulation/Tape Pull 335
FORCE OK 248	Order Extension
FREE 181	CTMAPI 892
GRAPHIC FLOW 122, 162	Order ID
GROUP 182	Multiple Orders 68
HOLD 180	ORDER Option
INSERT 113, 126, 257, 344, 355	CMEM Table List 255
IOA Primary Option 86	Job List Screen 116, 126
ISPF PACK 110	Manual Scheduling 151
ISPF Primary Option Menu 102	Table List Screen 116
. <i> </i>	

ORDERID Command	MEMNAME Parameter 572
Active Environment Screen 180	OVERLIB Parameter
ORDERID Field	AutoEdit Variable 796
Active Environment Screen 172	Description 571
IOA Log Show Screen 304	Example 576
Job Order Execution History 235	JCL Member Name 331
WHY Option 290	OVERRIDE DAILY PLAN Field
Zoom Screen 215	Parameter Prompting 356
Ordering Jobs 150	OVERRIDE DAILY PLAN Options
End User Job Order 370	FETCH A PLAN Screen 835
Job List Screen 126	Overwrite Confirmation
Job Ordering Utility 326	Quick Schedule Definition 364
Next Day 801	OWNER Field
Overview 66	Active Environment Screen 171
Ordering Rules	Job Order Execution History 235
CMEM Rule Table 267	Quick Schedule Definition 362
ORDERING Status	Show Screen Filter 197
Active Environment Screen 189	OWNER Parameter
Original Scheduling Date	AutoEdit Simulation 330
ODATE 63, 750	CMEM Rule 737
ORIGLIB Field	Description 577
Zoom Screen 215	DO RULE Statement 707
OS/390 Restart	Example 578
@ Character 819	
OUT Parameter 933	<b>D</b>
procstep.pgmstep 934	P
OUT Condition	
@#-procstep.pgmstep 934	P Option
Group Entity 566	Job List Screen 126
IOALDNRS Utility 817	PA1 Key 95
Job Dependency 242	AutoRefresh Mode 101
OUT Parameter	AutoRefresh Mode 101
DATEREF Subparameter 563	PA2 Key 95
Description 562	PAGE
DO COND Statement 440, 566	Scrolling Amount 95
Example 567	PAGES Field
Job Chain 569	Job Order Execution History 236
Logic 565	PANVALET
Long Condition Names 562, 565	Job Documentation 146
OPT Subparameter 563	PANVALET Product
OS/390 Restart 933	CONTROL-M 525, 572
Prerequisite Condition 501	PARAM PROMPTING Option
Quick Schedule Definition 361	Online Utilities Menu 323
OUT Statement	Parameter
Prerequisite Condition 70	#JNFRT 217
Output Class	AECACHL 758
SYSDATA 68	AUTOTAPE 241, 598
	CTGFORC 417
Sysout 646	DUEINCHK 493
Output Files	FRCOKOPT 247, 553
Simulation 841	GRPRECHK 112
Output Registers	IOADFLT 423
CTMAPI 883	
OVERICLM Parameter 536	MAXCCOK 552, 553 TAGMAXWT 520
OVERLIB Library	
COPMEM2O Parameter 572	Parameter Definition
Deleting JCL Member 572	Parameter Prompting 340, 342, 353
ICL Member 182 571	PARAMETER LIBRARY Parameter

AutoEdit Simulation 331	WDAYS Parameter 673
Parameter Passing	PDSE Library 579
MEMLIB Parameter 526	PDSE-type Library
Parameter Prompting	MINIMUM Parameter 531
AutoEdit Parameter 343, 347	PDSMAN
Daily Prompting Tables 345	\$\$\$SPACE Member 158, 267, 320
Daily Scheduling Table 831	PENDING Conditions
Daily Table 345, 828	Manual Conditions 292
Define Parameters Option 341	Periodic Calendar 316, 319
Definition Phase 829	DCAL Parameter 423
EXEC Phase 831	Example 424, 673
FETCH Phase 830	Overlapping 319
File Prefix 339	WCAL Parameter 672
IRS Tape Example 824	WDAYS Parameter 424, 673
Master Plan 351	periodic calendar
Master Table 340	STAT CAL parameter 639
New Method 826	STAT CAL PERIOD parameter 640
Old Method 824	statistics 639, 640
Scheduling Table 829	Periodic Scheduling
Type 1 339, 824	Format 423, 671
Type 2 349, 829	PF01/PF13
Parameter Update	HELP Command 93, 99
Parameter Prompting 340	PF02/PF14
Parameters	ISPF SPLIT Command 93
Basic Scheduling 131	SHOW Command 93
CMEM Rule Facility 259	SPLIT Command 101
Description 388, 696	PF03/PF15
General Job 130	CMEM Rule Definition 264
Group Scheduling 140	END Command 93
Job Scheduling 44	IOA Log Show Screen Window 306
Multiple Occurrences 129	Scheduling Definition 148
Postprocessing 135	Show Screen Filter 199
Runtime Scheduling 134	PF04/PF16
Scheduling Definition 128	Active Environment Screen 180, 199
Summary 377	Box Color 163
Tape Pull List 853	CMEM Rule Exit Option 265
PARM Field	Global View Screen 200
Parameter Prompting 343	Job List Exit Window 150
PARM NAME Field	REFRESH Command 200
Parameter Prompting 353, 359	RESET Command 93
PARM PREFIX Field	PF05/PF17
Parameter Prompting 343, 347, 354, 358	FIND Command 93, 97, 162
Update Parameter Values 359	PF06/PF18
Passing Arguments	=6 Command 93, 109
ARGUMENTS Parameter 413	PF07/PF19
DO CTBRULE Statement 442	Filtering 306
Password	Show Screen Filter 199
Online Facility 85	UP Command 93
PC PACKET STATUS Option	PF08/PF20
Primary Option Menu 88	DOWN Command 93
PDS Parameter	Filtering 306
CONFCAL Parameter 404	Show Screen Filter 199
DATES Parameter 418	PF09/PF21
DAYS Parameter 424	SWAP Command 101
Description 579	PF10/PF22
MINIMUM Parameter 531	IOA Log Show Screen Window 306
MONTHS Parameter 533	LEFT Command 93
arailletter voo	LLI I Communa vo

PREV Command 93	Job List Screen 126, 164
Scheduling Definition 149	PLAN ORDERED ALREADY Field
Show Screen Filter 199	Parameter Prompting 357, 358
PF11/PF23	Plan Selection Screen
Active Environment Screen 179	Parameter Prompting 357, 359
NEXT Command 93	PLANID Parameter
RIGHT Command 93	CTMEXEC CLIST 834
Scheduling Definition 149	CTMFETCH CLIST 833
PF12	Daily Scheduling Table 831
RETRIEVE Command 94	PLANID Suffix
PF24	CTMFETCH CLIST 833
SHPF Command 94	POOL DEFINITION Option
PFKey Definition	Primary Option Menu 89
Online Facility 93	Post-Processing
PRINT Command 101	System Variable 754
TSO CTMTTRA 109	post-processing
PFKey Display	statistics 639
SHPF Command 94	
PFKey functions	Post-Processing Parameter
IOA Editor 103	System Variable 753
	Post-Processing Parameters
PFKeys	CONTROL-M Monitor 50
DOWN 95	DO SET Statement 467, 623
UP 95	Group Entity 141
PGMST Parameter	Scheduling Definition 135
ON Statement 543	Summary 385
Step Range 547	Post-processing Statement
PGMSTEP Parameter	Error 386
ON DSNEVENT Statement 725	PPF2DAY Job
ON STEP Statement 734	JCL Library 838
pgmstep.procstep	PPF2DEL Job
DO IFRERUN Statement 448	JCL Library 838
Pipe	Precedence
Job Scheduling Definition 57	AutoEdit Variable Assignment 769
Job-Related Considerations 821	Predecessor Job
MAINVIEW Batch Optimizer (MVBO) 56	Job Dependency 75, 242
PIPE Field	REFRESH Command 245
Show Screen Filter 197	Prefix
PIPE Parameter	Maybe Job Prerequisite Condition 81
Description 581	PREFIX Field
MAINVIEW Batch Optimizer (MVBO) 375, 385	Manual Conditions 284, 292
MAINVIEW Batch Optimizer (MVBO)	Prefixing
Implementation 821	Description 83
Summary 134	PREREQ CONDITION Option
Pipe Participant	Online Utilities Menu 323
Definition 57	Prerequisite Condition
PIPE Statement	Add/Check/Delete Utility 324
Deleted Through Zoom Screen 821	Adding 285, 293, 563
PLAN Command	CMEM Rule 681
Scheduling Definition 142	CONTROL-M Files Prefix 340
Plan Description	Date Reference 71
Parameter Prompting 351	Deleting 72, 287, 294, 564
PLAN NAME	Description 69
Master Prompting Plan PREFIX 357	DO COND Statement 436, 701
Parameter Prompting 356, 359	DO Statement 434
PLAN NAME Prefix	Erasing 287, 294
Parameter Prompting 351	Example 71, 506, 568
PLAN Option	Format 361
· · · · · · · · · · · · · · · · ·	

Group Entity 71, 566	PRIORITY Parameter
IMSACTIVE 569	Description 588
IN Parameter 500	Example 589
IOALDNRS Utility 817	Job Flow 75
IRSTAPEARRIVED 824	Logic 589
Job Dependency 71, 75	PRM Parameter
Manual Conditions 60, 291	DO SYSOUT Statement 477
Manual Intervention 72, 502, 564	PRMTBLB Parameter
Maybe Job 819	CTMFETCH CLIST 833
OUT Parameter 562	PRO/JCL
Parameter Prompting 340, 343	INVOKE JOBSCAN Parameters 855
Quick Schedule Definition 363, 365	Simulation/Tape Pull 333
Runtime Criteria 48	
RUN%PLANID 831, 835	Tape Pull List 855
•	PROBLEMS READING SYSOUT Status
Show Screen Filter 197	Active Environment Screen 187
STAT 72	PROCST Parameter
Unscheduled Condition 819	+EVERY 544
Why Screen 208	ON Statement 544
Prerequisite Conditions	PROCSTEP Parameter
IOA Conditions File 281	ON DSNEVENT Statement 725
IOA Conditions/Resources Screen 281	ON STEP Statement 733
PREV Command	PROD Mode
Job Scheduling Plan 165	CMEM Rule Simulation 719
PF10/PF22 93	Product Description
Scheduling Definition 143, 149	CONTROL-M/Restart 42
PREV Value	CONTROL-M/Tape 43
FIND Command 97	CONTROL-O 43
IN Parameter 501	CONTROL-O/COSMOS 43
Schedule Date 437, 563	CONTROL-D 43
prevent NOT CATLGD2 errors 585	CONTROL-D/Image 43
PREVENT-NCT2 Parameter	CONTROL-D/Page On Demand 43
Description 584	CONTROL-M 42
PREVENTNCT2 Parameter	CONTROL-V 43
Example 587	CONTROLM/Analyzer 43
Previous	product support 3
SAC Parameter 610	Production Delay
PREVYEAR Command	MAXWAIT Parameter 519
Calendar Definition Screen 321	Productivity Tools
Print a Copy of the Screen 95	Option OA 89
PRINT Command	Profile Variable
PFKey Definition 101	SACTMOD 168
Print Screen 95	Programmer Information
Printing	SHOUT Parameter 629
8	PROMPT IND Field
Sysout 479, 646, 649 PRIOR RUN Status	
Active Environment Screen 187	Parameter Prompting 353
	PROMPT Library
Priority	%%LIBSYM Statement 828
Conditions/Resources 283, 290	Prompting Plan
Overview 74	AutoEdit Variables 824, 830
Runtime Criteria 47	PROPAGATE Argument
SYSOUT Operations 481	REFRESH Command 245
Sysout Operations 649	PRTDBG
User-Defined Variable 765	DD Statement 95
PRIORITY Field	PRTY Field
Show Screen Filter 197	Job Dependency Network 244
Priority Field	PSEUDO Job
WHY Option 290	Status 389

PSEUDO Jobs	Active Environment Screen 181
MEMBLIB Parameter 390	RECAPTURE ABEND CODE
	ON Statement 546
	STEP RANGE Parameter 643
Q	RECAPTURE CONDITION CODE
4	ON Statement 546
Quantitative Resource	STEP RANGE Parameter 643
Adding 285	RECAPTURE CONDITION CODES Field
Changing 288	Restart Confirmation Window 226
Critical Path Priority 589	RECIPIENT TREE Option
Definition 73	Primary Option Menu 88
Deleting 287	Record Selection Criteria
MAINVIEW Batch Optimizer (MVBO)	Active Environment Screen 189
Implementation 822	REFRESH Command
RESOURCE Parameter 596	Active Environment Screen 176
Runtime Criteria 48	Global View Screen 200
Show Screen Filter 197	Job Dependency Network 242, 244
Quantitative Resource Return Codes, CTMAPI 900	RELATIONSHIP Field
Quantitative Resources	
IOA Conditions/Resources Screen 281	Group Scheduling Table 375
QUANTITY Field	RELATIONSHIP Parameter
Conditions/Resources 283	Description 127, 591
	Group Scheduling 128
WHY Option 290	Group Scheduling Logic 383
Quick Schedule Definition	Group Scheduling Table 379
CTMQUICK Utility 360	RELEASED Status
Example 369	Active Environment Screen 188
Exiting 368	REMAIN Parameter
Overwrite Confirmation Window 364	CTMEXEC CLIST 834
Screen 361	REMAINING PARAMETERS Field
QUICK SCHEDULE Option	Parameter Prompting 357
Online Utilities Menu 323	Update Parameter Values 359
Quick Submit Command	Remedy Helpdesk closing 461
Job Ordering 813	Remedy problem ticket
QUICKDEF Utility	closing ticket 461
ISPF Online Utility 377	opening 460
	Remote Node
	NJE Network 51
R	REMOVE UNSCHED CONDITIONS
	Group Entity 820
R Option	REP3LEFT Report
Active Environment Screen 181, 222	Simulation/Tape Pull 336
Parameter Prompting 344	REPBYDSN Parameter
RACF	Tape Pull List 853
Security Product 577	REPBYJOB Parameter
RANGE Control Statement	Tape Pull List 853
%%RANGE 777	REPBYTIME Parameter
RBA 175	Tape Pull List 853
RBA Field	REPBYVOL Parameter
Active Environment Screen 175	Tape Pull List 853
Conditions/Resources 175, 284	
WHY Option 290	Repeat Commands
RBAL Command	Edit Environment 916, 929
Active Environment Screen 175, 284	REPEAT Option
Job Dependency Network 242	Parameter Prompting 344, 354
RDR=INTRDR Parameter	REPLACE Parameter
CTMAESIM Utility 794	CTMFETCH CLIST 833
Reactivate Option	REPLACE YES Option
weather option	CTMFETCH CLIST 835

Replies	DO Statement 434
CTMBAPI 902	RERUN Status
REPORT BY DSN Field	Show Screen Filter 195
Simulation/Tape Pull 336	RERUN Value
REPORT BY JOB Field	SHOUT Parameter 628
Simulation/Tape Pull 336	Rerun/Restart Window
REPORT BY TIME Field	\$FIRST/\$CLEANUP Values 228
Simulation/Tape Pull 336	Active Environment Screen 224
REPORT BY VOLSER Field	RERUNMEM Parameter
Simulation/Tape Pull 336	Description 593
Report Decollating	MAXRERUN Parameter 517
D-CAT Parameter 416	RES Field
REPORT DEF Option	Conditions/Resources 285
Primary Option Menu 88	Job Dependency Network 244
Reporting Facility	RES NAME Field
Overview 55	Show Screen Filter 197
REPORTS Field	RESET Command
Simulation/Tape Pull 336	CMEM Rule Exit Option 265
Repository	Description 98
Description 62	Edit Environment 914
REQUESTED CHANGE HELD Status	Exit Option 150
Zoom Screen 222	IOA Log Show Screen Window 306
REQUESTED CHANGE Status	PF04/PF16 93, 98
Active Environment Screen 188	Quick Schedule Definition 369
REQUESTED DELETE	Scheduling Facility Exit Option 150
Active Environment Screen 189	Show Screen Filter 199
REQUESTED FORCE OK Status	RESOLVE Control Statement
Active Environment Screen 188	%%RESOLVE 778
REQUESTED FREE Status	Resource
Active Environment Screen 188	Conditions/Resources 175
FREE Option 181	Resource Allocation
REQUESTED HELD Status	Critical Path 74
Active Environment Screen 188	Resource Contention
HOLD Option 180	Critical Path Priority 589
REQUESTED REACT Status	Resource Control
Active Environment Screen 188	CONTROL Parameter 408
REQUESTED RERUN Status	RESOURCE Parameter
Active Environment Screen 188	Automatic Tape Adjustment Facility 241
Rerun	Description 596
Definition 67	Example 600
DO STOPCYCL 475	Logic 598
Rerun Confirmation Window	RESOURCE TYPE Field
CONTROL-M/Restart 225	Show Screen Filter 197
Manual Job Rerun 222	Resource Utilization
Rerun Interval	Critical Path 74, 588
INTERVAL Parameter 513	Priority 588
Rerun Job	Tape Devices 241
Example 795	Resource Window
RERUN NEEDED Status	Conditions/Resources 288
MAXRERUN Parameter 516	Resources
RERUNMEM Parameter 593	Forecasting 840
RERUN Option	Simulation 840
Active Environment Screen 181, 222	Resources File 62
TASKTYPE Parameter 655	RESTART
RERUN Parameter	ON Statement 546
Example 517	STEP RANGE Parameter 643
Rerun Request	Restart
The state of the s	Ivottui t

CONTROL-M/Restart 67	Rule Definition
Definition 67	Editing 927
DO IFRERUN Statement 447	Maintaining Validity 930
DO STOPCYCL 475	RULE DEFINITION Option
RESTART DECISION Field	Primary Option Menu 89
Zoom Screen 216, 218	RULE Field
Restart Job	CMEM Entry Panel 253
DO IFRERUN Statement 447	Rule List Screen
OUT Parameter 935	Commands 256
Restart OS/390	Copying Rules 269
OUT Parameter 933, 935	RULE STATUS Option
RESTART Parameter	IOA Menu 89
DO Statement 434	Rule Table
Restart Step	Automatic Creation 361
List Window 233	CMEM 250
OUT Parameter 934	Rule, CONTROL-M/Analyzer
RESTARTED Status	CTB STEP Parameter 413
Active Environment Screen 188	RULENAME Parameter
Restoration	DO RULE Statement 707
Active Jobs File 49	RUN Attribute
Conditions/Resources 49	ODATE 65
RESTORE Option	RUN n Status
History Environment Screen 247	Active Environment Screen 188
RESTORED Status	RUN SIMULATION Field
Active Environment Screen 188	Simulation/Tape Pull 334
RETENTION Parameter 603	Run Statistics
# OF DAYS TO KEEP 603, 605	Statistics Screen 238
	Runtime Criteria
# OF GENERATIONS TO KEEP 603, 605	
History Jobs File 128, 375	Job Submission 47
Retention Period	Runtime Scheduling
SYSDATA 399	MAINVIEW Batch Optimizer (MVBO) 822
Retrieval Criteria	Pipe Algorithm 57
Selection Criteria 284	Runtime Scheduling Parameter
RETRIEVE Command	THRESHOLD 741
PF12 94	Runtime Scheduling Parameters
RETRO Parameter	Scheduling Definition 134
Description 607	Summary 385
MAXWAIT Parameter 520	RUNTSEC Parameter
MINIMUM Parameter 531	Example 740
PDS Parameter 579	Security Check 740
Return Codes	RUN%%PLANID
CTMAPI Forcing Jobs 867	Prerequisite Condition 831, 835
CTMAPI Ordering Jobs 867	
RIGHT Command	C
PF11/PF23 93	5
ROSCOE	
DO SHOUT Statement 712	S Option
ROSCOE/ETSO	Active Environment Screen 181
Address Space 81	Display Filters Window 301
PF06/PF18 93	End User Job Order 370
Row commands	Job List Screen 125
IOA Editor 104	Table List Screen 121
Row Numbering	S*37 Abend Code
Variable Database Facility 277	SET VAR Parameter 624
RULE ACTIVITY Option	SAC Parameter
Primary Option Menu 89	+ (Group Next) 610
Rule Copying 269	- (Group Previous) 610

Next 610	DO MAIL Statement 452
Previous 610	DO REMEDY Statement 460
SAMPEXIT Library	Example 375
CTMX002 Exit 525	General Parameters 377
SAP/R3	Logic 382, 421
architecture 908	Non-periodic 670
BAPI 909	Periodic 423, 671
running on USS 908	Runtime Parameters 385
Unix and the SAP/R3 Application Layer 910	TASKTYPE Parameter 654
with DB/2 database 909	Scheduling Criteria
SAVE (Y/N) Field	Group Entity 111
IOA Log Show Screen 302	Job Submission 47
Show Screen Filter 193	Scheduling Definition
SAVE Command	Commands 142
Zoom Screen 219, 221	Creation 113
SAVE DOCUMENTATION Parameter	Deletion 115, 158
Save Documentation Window 147	Description 110
Save Documentation Window  Save Documentation Window	Documentation 144
Scheduling Definition 146	Editing 143, 913
SAVE Field	Entry Panel 116
Exit Option Window 150	Exiting 148
IOA Log Show Screen Window 302	Graphic Jobflow 162
Scale Line	Group Entity 138
View Graph Screen 203, 204	
SCATMOD Profile Variable 168	Group-Handled Jobs 112 Job List 123
SCHDLIB Field	
	Job Plan 164
Zoom Screen 215	Job Scheduling 375
SCHDTAB Field	Ordering Jobs 151
Zoom Screen 215	Overview 58
Schedule Date	Parameters 128
Job Scheduling Plan Screen 166	Screen 110, 127
OUT Parameter 563	Search Window 119
SCHEDULE PREVIOUS DAY Parameter	Table List 120
Description 610	Scheduling Definition Screen
SCHEDULE TAG ACTIVE Parameter	Group Entity 139
Description 616	Scheduling Information
Format 616	Job Scheduling Plan Screen 165
FROM 133, 616	Scheduling Jobs 150
SCHEDULE TAGS, Conflicting 618	Scheduling Library
UNTIL 133, 616	DO FORCEJOB Statement 444
SCHEDULE TAG Field	Job Scheduling 44
Group Scheduling Table 375	Scheduling Library Mode
SCHEDULE TAG Parameter	AutoEdit Syntax Testing 329, 793
Description 612	Parameter 331
Group Entity 111, 141	SCHEDULING LIBRARY Parameter
Group Scheduled Job 501	AutoEdit Simulation 331
Group Scheduling 128, 131, 379	Description 327
SCHEDULE-PREV-DAY Value	Scheduling Logic
DESC Parameter 432	DAYS Parameter 421
SCHEDULE-PREV-ONLY Value	Description 382
DESC Parameter 432	Scheduling Parameters
SCHEDULED RUN DATE Parameter 327	Display 117
Scheduling	Scheduling Table
Basic Parameters 379	DO FORCEJOB Statement 444
Calendar Facility 306, 315	Job Scheduling 45
Description 375	Parameter Prompting 829
DO FORCEJOB Statement 444, 445	SCHENV parameter

format 619	Job Dependency Network 243
SCHTBLB Parameter	Job List 123, 151, 164, 365, 370
CTMFETCH CLIST 833	Job Log 212
Screen	Job Order Execution History 234
Printing 95	Job Scheduling Definition 127, 145, 165, 369, 913
Screen Control	job scheduling definition 375, 393, 400, 407, 410, 417
Online Facility 109	419, 432, 443, 446, 451, 468, 474, 483, 489, 491, 493
Screen Description Line	497, 499, 506, 515, 518, 521, 527, 530, 532, 556, 567
Online Facility 92	576, 578, 580, 587, 600, 625, 636, 637, 643, 651, 657
Screen Exit	Jobflow 162
CANCEL Command 99	Manual Conditions 291, 293, 294
Screen Help	Master Plan Definition 351
Line Sensitive 99	Master Table Definition 341
Online Facility 99	Online Utilities 323
Screen Layout	Online Utilities Menu 323
Online Facility 92	Parameter Prompting Facility 341
Screen Printing	Parameter Prompting Type 1 Menu 339
ISPF LIST File 101	Parameter Prompting Type 2 Menu 349
Screen Transfer	PFKey Window 94
TSO CTMTTRA Command 109	Plan Selection 357
Screens	Prerequisite Condition Utility 325
Active Environment 167	Quick Schedule Definition 361
AutoEdit Simulation 329	Rule Definition Entry Panel 117
Calendar Definition 316	Scheduling Analysis 205
Calendar Facility 307	Scheduling Definition 110
Calendar Facility Entry Panel 308	Scheduling Group 138
Calendar List 309	Set Conditions 347
CMEM Online Entry Panel 252	Simulation, AutoEdit 328
CMEM RULE Definition 709	Simulation, CONTROL-M 332
CMEM Rule Definition 258, 263, 693, 697, 714, 716,	Statistics 238
720, 728, 730, 732, 740, 927	Sysout Viewing 236
CMEM Rule Facility 250	Table List 120, 158
CMEM Rule List 255	Table Selection 345
CMEM Table List 254	Tape Pull List 332, 855
Conditions/Resources 286, 287	TSO Command Processor 108
CONTROL-M Simulation 332	Update Parameters 347, 358
Database List Screen 273	Variable List Screen 274
Define or Update a Master Plan 351	Variable Zoom Screen 278
Define Parameters and Conditions 341, 342	View Graph 202
Define Parameters in Master Plan 352	Why 205
Edit Environment 143, 263, 913, 927	Year List 310
Entry Panel 116	Zoom 215
Exec/Order a Plan 356	SCROLL Field
Fetch a Plan 355	Screen Header 95
Forecasting Facility 332	Scrolling
Global View 199	Commands 95
Graphic Jobflow 162	SEARCH COUNTER Field
Group Scheduling 138	DISAPPEARED Status 217
History Environment 245	Zoom Screen 217
IOA Entry Panel 84	Search Function
IOA Help 99	CTMAPI 872
IOA Log 94, 296	Searching
IOA Log Show Screen Window 301, 304	FIND Command 96
IOA Primary Option Menu 85, 87	LOCATE Command 96
IOA TSO Command Processor 108	Security
IOA Variables Facility 272	CMEM Rule 737
Issue a Job Request 326	OWNER Parameter 577
1	

RUNTSEC Parameter 739	Setting Variable Values
Security Exit IOASE07 900	AutoEdit 779
SELECT Command	Shared Control
Table Creation 112	CONTROL Parameter 408
Table List Screen 122	Shared Resource
SELECT Option	WAIT SCHEDULE Status 286
Calendar List Screen 310	SHIFT Parameter
CMEM Rule List 257	CONFCAL Calendar 381
CMEM Table List 255	CONFCAL Parameter 403
Display Filters Window 301	SHOUT Facility
Display Filters window 191	CONTROL -O 52
Job List 125	Shout Facility
Table List 114, 121	DO SHOUT Statement 469
Year List Screen 312	Problem Notification 48
Select Option	SHOUT Message
Job Order Execution History 236	AutoEdit Variable 796
Selection Criteria	IOA Log Show Screen Window 303
Active Environment Screen 189	SHOUT Parameter
CMEM Actions 259	Description 627
Conditions/Resources 284	DO SHOUT Statement 474, 635
Display Filters window 191	Job Statistics File 631
Parameter Prompting 343	System Variables 747
Show Screen Filter 193	SHOUT Statement
SELECTION FIELD	%%JOBNAM/%%JOBNAME Variables 361
Parameter Prompting 359	Post-Processing Parameters 136
SET	SHOUT WHEN EXECTIME Message
Global Variables 271	Job Dependency Network 244
SET command 105	SHOUT WHEN LATE Message
SET Command Panel	Job Dependency Network 244
End TRACE level 106, 107	New Day Procedure 632
Set Dollar Sign Representation 107	SHOUT WHEN LATESUB Message
Set TRACE level 106, 107	Job Dependency Network 244
Set Conditions	New Day Procedure 632
Screen 343	SHOW Command
SET Control Statement	Active Environment Screen 175, 190, 300
%%SET 780	IOA Log Screen 298
Set Dollar Sign Representation	Log Screen 213, 299
SET Command Panel 107	PF02/PF14 93
SET Statement	SHOW JOB DOCUMENTATION Field
JCL SET 770	Entry Panel 144
Set TRACE level	SHOW LIMIT ON Field
SET Command Panel 106, 107	Log Screen 297
SET VAR	Show Option Window 189
Global Variables 271	Show Screen Filter
SET VAR Parameter	Activating 190, 192
AutoEdit Variable 466, 622	Active Environment Screen 190, 192
Description 621	Displaying available filters 190
DO SET Statement 467, 623	Exiting 198
Example 623	Fields 193
SET VAR Statement	Show Screen Filter Window Field
AutoEdit Statement 795	DESC 193
Global Variable 760	SHPF
JCL Setup 765	Command 262
Local Variable 757	SHPF Command
User-Defined Variables 747	CMEM Rule Definition 262
SETOGLB	PF24 94
Global Variables 271	Show PFKey 94

SIMEND Parameter	+EVERY Step 549
Simulation 843	ON Parameter 554
SIMLOG Output File	SORT Command
Simulation 846	Job List Screen 125
SIMOAJF Output File	Space Allocation
Simulation 846	SET VAR Parameter 623
SIMOCND Output File	Space Report Field
Simulation 846	Simulation/Tape Pull 337
SIMORES Output File	SPD Statement 610
Simulation 847	Special Catalog
SIMPARM DD statement	Tape Pull List 852
CONTROL-M 843	Special Purpose Job
SIMSTART Parameter	Job Ordering 815
Simulation 843	Job Scheduling 812
SIMUL/TAPE PULL Option	SPLIT Command
Online Utilities Menu 323	PF02/PF14 93, 101
Simulation	Split Screen Mode
Active Jobs File 840, 845	Online Facility 101
Analyzing the Run 848	SRB Time Field
AutoEdit Statement 793	Statistics Screen 240
CMEM Rules 719	SRB Time, Group
CTMAESIM Utility 793	Statistics Screen 240
CTMCSIM CLIST 332	SSCHTBO Parameter 156
CTMX002 847	START Command
Description 840	TASKTYPE Parameter 655
Input Files 841, 845	START Field
INVOKE JOBSCAN 332	Job Order Execution History 236
Manual Conditions 850	START TIME Field
Message File 846	Statistics Screen 240
MODE Parameter 719	STARTED Status
Output Files 846	Active Environment Screen 188
Overview 54	Started Task
Parameters 843	AUTOARCHIVE Parameter 399
Screens 332	MEMLIB Parameter 524, 527
Tape Pull List 851	MEMNAME Parameter 528
Simulation Facility	OVERLIB Parameter 572
CTMCSIM Utility 332	Show Screen Filter 196
Simulation Option	TASKTYPE Parameter 654
Active Environment Screen 183	STAT CAL Parameter
SIMULATION Parameter	Description 639
Simulation/Tape Pull 335	STAT CAL PERIOD Field
Simulation/Tape Pull Utility	Zoom Screen 215
CLIST CTMCSIM 332	STAT CAL PERIOD parameter
SKELETON Field	description 640
Quick Schedule Definition 362	STAT Command
Skeleton Job	Job List Screen 124, 312
Quick Schedule Definition 360	Rule List Screen 257
SMFID Parameter	STAT Date Reference
ON DSNEVENT Statement 724	DO COND Parameter, DATEREF Subparameter 437
ON JOBARRIV Statement 729	DO COND Statement 700
ON JOBEND Statement 731	IN Parameter 501
ON STEP Statement 733	OUT Parameter, DATEREF Subparameter 563
SMS Support	Prerequisite Condition 72
CMEM 691	STAT Field
SNRUN	Conditions/Resources 285
ANYSTEP 554	Global view Screen 201
SNRUN Code	Manual Conditions 293

STAT Message Type	Show Screen Filter 196
IOA Log Show Screen Window 303	STEP ADJUSTMENT Field
STAT Option	Restart Confirmation Window 227
Active Environment Screen 181	Step Event
Job List Screen 126	CMEM 680
State (Mode of Control)	DO FORCEJOB Statement 704
CONTROL Parameter 408	Step List Window
STATE Status	Restart Window 233
Show Screen Filter 196	Step Range
Statement	Example 558
DO SET Statement 765	ON Statement 543
Error	PGMST Parameter 543, 547
Post-processing 386	STEP RANGE Parameter
Statistical Information	Description 641
Global View Screen 199	Example 643
Job Status 176	pgmstep.procstep 641
View Graph Screen 201	STEP Statement 259
Statistics	STEPRC field
Execution Information 54	DSNEVENT criteria 689
Group Entity 240	STEPRC Parameter
JOBSTAT Command 142	
	ON DSNEVENT Statement 726, 727 ON STEP Statement 734
Tape Device 241 statistics	
	String Search LOCATE Command 96
STAT CAL PEDIOD	SUBMIT Command
STAT CAL PERIOD parameter 640	
utility 639	From ISPF 813
Statistics File	ISPF 813
SHOUT Parameter 631	Quick Submit 813
Statistics Screen 238	SUBMIT Function
Active Environment Screen 176	AutoEdit Simulation 331
Fields 239	SUBMITTED Status
JOBSTAT Command 176	Active Environment Screen 188
STATUS Field	Show Screen Filter 195
Active Environment Screen 172	SUBSCAN Function
Job Dependency Network 244	AutoEdit Simulation 331
Job Order Execution History 236	Subtraction Operation
Show Screen Filter 194	%%MINUS 781
Status Reply DSECT	Successor Job
CTMBAPI 889	Job Dependency 75, 242
Status Return Codes 891	SUFFIX Field
Status Returned	Quick Schedule Definition 364
CTMBAPI 890	SUFFIX Parameter
Status Screen 166	CTMEXEC CLIST 834
Functions 59	CTMFETCH CLIST 833
Manual Confirmation 449	SUM Field
WAIT CONFIRMATION Status 406	View Graph Screen 203
Status screen 462	SUMM
Status Zoom Screen	DO REMEDY parameter 460
DOC Lines 490	support, customer 3
Status, CONTROL-M	SWAP Command
Job Dependency Network 243	PF09/PF21 101
STATUS, Group	Sxxx Code
Statistics Screen 240	+EVERY Step 549
STC	Syntax Checking
Started Task 523	AutoEdit Statement 793, 851
STC Task Type	CTMSCIM Utility 333
IOA Log Show Screen Window 304	Edit Environment 914

JCL Statements 851	System Abend Code
SYSDATA	ON Statement 550
Definition 68	System Date
SYSDATA Archiving	Definition 63
AUTOARCHIVE Parameter 398	DO FORCEJOB Statement 444
SYSDATA Deletion	
	JCL Setup 750 SYSTEM ID Parameter
Active Jobs File 399	
SYSDATA Viewing	Format 652
AUTOARCHIVE Parameter 398	Under JES2 652
SYSDB Parameter	Under JES3 653
AUTOARCHIVE Parameter 398	System Messages
SYSIN DD Statement	SYSDATA 68
%% Parameter 807	SYSTEM Parameter
SYSOPT = CTMWORK	ON DSNEVENT Statement 724
CONTROL-M/Analyzer System Variable 443	ON JOBARRIV Statement 729
SYSOUT Archiving	ON JOBEND Statement 731
Option F 796	ON STEP Statement 733
SYSOUT Dataset	SYSTEM Subparameter
SYSDATA 68	DO SHOUT Statement 711
Sysout Destination	System Variable
DO SYSOUT Statement 480	AutoEdit 745, 748
Sysout destination	Date Variable 750
SYSOUT Parameter 647	JCL Setup 748
SYSOUT Operations	MEMLIB Parameter 796
Copying 646	Post-Processing 753, 754
SYSOUT Parameter 645	Resolution 765
SYSOUT operations	SET VAR Parameter 466, 622
Priority 481	SHOUT Parameter 747
Sysout Operations	SYSOPT = CTMWORK 443
Archiving Facility 484	SYSOUT Parameter 747
Class Change 479, 647	
Copying 478, 482, 649	
Displaying 236	T
DO SYSOUT Statement 478	•
Merging 481, 648	T Option
Moving 480, 647	Active Environment Screen 181
Multiple 480, 647	Job List Screen 126
Printing 479, 647, 649	Table
Priority 649	Browse Mode 114
•	Creation 112, 122, 250
Releasing 479, 646 SYSDATA Definition 68	Deletion 115, 158
	TABLE Command
Viewing Screen 236	Active Environment Screen 177
Sysout Option Code	Table Description
DO SYSOUT Statement 477	Parameter Prompting 341
SYSOUT Parameter 644	TABLE Field
SYSOUT Option F	Active Environment Screen 172
AutoEdit Variable 796	
SYSOUT Parameter	CMEM Entry Panel 253
Description 644	Quick Schedule Definition 362
DO SYSOUT Statement 483, 650	Scheduling Facility Exit Option 14
Example 651	Table Information
Logic 647	Quick Schedule Definition 361
System Variables 747	Table Library
SYSPLEX	Parameter Prompting 341
Variable Database Facility 271	Table List
Sysplex	CMEM Table List 120
Minus-One Support 55	Options 121

Statistical Information 120	IOA Log Show Screen Window 304
Table List Screen	TASK TYPE Field
CMEM Rule Facility 254	Show Screen Filter 196
Commands 122	taskid Format
Delete Confirmation 158	MEMLIB Parameter 524
Description 120	TASKTYPE Parameter
Exiting 150	Description 654
Job Ordering 813	Example 657
Manual Job Scheduling 150	MEMLIB Parameter 524
New Table 150	MEMNAME Parameter 529
Options 114	Tasktype WRN
TABLE NAME Parameter	Warning Message 656
AutoEdit Simulation 331	technical support 3
Description 327	Terminal Support
Entry Panel 114	Online Facility 82
Table Name Prefix	TERMINATE Option
Parameter Prompting 341, 828	Active Environment Screen 181
TABLE Parameter	TEST Mode
DO FORCEJOB Statement 444, 703	CMEM Rule Simulation 719
DO RULE Statement 707	Testing
TABLE PREFIX Field	AutoEdit Syntax 793
Table Selection Screen 346	THRESHOLD Parameter
Table Selection Screen	Description 741
Parameter Prompting 345	THRESHOLD Runtime Scheduling Parameter 741
TAGMAXWT Parameter	Time
CTMPARM 520	Runtime Criteria 47
Tape Adjustment 55	TIME + DAYS Parameter
Tape Devices	Description 659
Statistics 241	FROM DAYS 659
Tape Drive	FROM TIME 659
RESOURCE Parameter 600	Runtime 659
	UNTIL DAYS 659
Tape Pull List	UNTIL TIME 659
Catalogs 852	
CTMCSIM Utility 332	Time and day limits TIME + DAYS Parameter 659
DD Statements 855	
JOB/SCAN, PRO/JCL, DOCU/TEXT 855	TIME Field
Parameters 853	Log Screen 297
Problem Determination 856	TIME Parameter
Recommendations 851	CTMEXEC CLIST 834
Sample Report 857	Description 659
Simulation 851	Runtime 659
Work Flow 851	TO DATE Field
TAPE PULL LIST Field	Date Range Window 164
Simulation/Tape Pull 336	TO Destination
TAPE PULL LIST Parameter	DO SHOUT Statement 470, 471
Simulation 854	SHOUT Parameter 629
Tape Pull List Parameters 853	TO Field
TAPULIN DD Statement	Zoom Screen 216
Simulation Parameter 853	TO Parameter
Tape Pull List 855	DO SHOUT Statement 710
TAPULOUT DD Statement	STEP RANGE Parameter 641
Tape Pull List 855	TO Step
TAPULPRM Member	DO IFRERUN Statement 449
CONTROL-M 853	Restart Step List Window 234
Target Computer	TO STEP Field
Example 804, 805	Restart Confirmation Window 226
TASK TYPE Criteria	TO=OPER Value

DO 01101/IED	TIN IDE A
DO SHOUT Destination 471, 472, 632, 633	TYPE 2
TO=TSO-ID Value	Parameter Prompting 338
DO SHOUT Destination 472, 633, 713	TYPE Field
TO=USERID Value	Conditions/Resources 283
DO SHOUT Destination 471, 632	Manual Conditions 292
Tocol Parameter	Parameter Prompting 354
FIND Command 97	Rule List Screen 255
TOP Command	Type of Task
Scrolling 96	TASKTYPE Parameter 654
TOP SECRET 577	TYPE Parameter
Totals Line	CTB STEP Parameter 413
Global View Screen 200, 202, 204	TYPRUN Parameter
Tracking and Control	Tape Pull List 854
Description 58	TYPRUN=HOLD
Tracking and Control Facility	CMEM On Spool Job 684, 688
Online Facility 166	TYPRUN=SCAN Parameter
Transfer Command	AutoEdit Simulation 331
Multi-Screen Control 90	Tape Pull List 852
Transfer of Control	Tupe I un Elst ook
TSO/Online Facility 93, 109	
Transfer to TSO/Utilities	U
PF06/PF18 93	U
TSO	U-M 472, 633
	U1 Option
AutoRefresh Mode 100, 176	Online Utilities Menu 372
Command Processor Screen 108	
Commands 108	UNDELETE Option
Control 93, 109	Active Environment Screen 182
Screen 108	UNEXPECTED CC Status
TSO Application	Show Screen Filter 195
Online Facility 109	UNITDEF Member
TSO CLIST facility 813	IOA PARM Library 241
TSO Cross Memory Option	Unix system services 905
PF06/PF18 93	Unnnn Code
TSO CTMTTRA Command	+EVERY Step 549
Transfer to IOA 109	Unscheduled Condition
TSO Environment	Manual Conditions 818
OWNER Parameter 577	UNTIL DAYS
TSO Job	TIME + DAYS Parameter 659
	UNTIL TIME
CMEM On Spool Job 683	TIME + DAYS Parameter 659
DO STOPJOB Statement 715	UNTIL Time
TSO Option	
Primary Option Menu 86	TIME Parameter 659
TSO SEND Command	Unused Tracks
DO SHOUT Statement 470, 711	MINIMUM Parameter 531
SHOUT Parameter 630	PDS Parameter 579
TSO Transfer Command	UP Command
PF06/PF18 93	PF07/PF19 93, 95
TSO User ID	UPDATE Option
Quick Schedule Definition 362	Parameter Prompting 344, 355
TSO-id Value	Update Parameters
DO SHOUT Destination 470	Fields 347, 359
TSO/ISPF	Screen 347, 358
	Type 1 Option 1 345
Environment 81, 126, 181	Updating Variables
TYP Field	Variable Zoom Screen 278
Active Environment Screen 172	UPP
TYPE 1	
Parameter Prompting 338	User Prompting Plan 831

UPPTBLB Parameter	CONTROL-M implementation 905
CTMEXEC CLIST 834	USS services 905
CTMFETCH CLIST 833	USS, CONTROL-M and
URGENCY	architecture for Unix-oriented implementation 907
DO REMEDY parameter 460	CONTROL-M Option for SAP/R3 909
URGENCY Field	in the USS environment 909
IOA Log Show Screen Window 303	JCL for OS/390 orientation 906
URGENCY Parameter	OS/390 oriented 906
DO SHOUT Statement 471, 473	SAP R/3 on USS 908
URGENCY Subparameter	SAP/R3 Application Layer 910
DO SHOUT Statement 711	Unix oriented implementation 907
URGN Parameter	Utilities
SHOUT Parameter 631	CLIST IOAUTIL 323
SHOUT Statement 634	Conditions/Resources File 325
Usage Line	CTMAESIM 328, 793
Screen Layout 92	CTMJBINT 370
USE Field	CTMJOBRQ 326, 813
Conditions/Resources 283	CTMJSA 238
WHY Option 290	CTMQUICK 360
User Abend Code	CTMSIM 840
ON Statement 550	CTMTAPUL 840
USER DATA Field	Fast Transfer 324
Statistics Screen 240	IOALDNRS 291, 813, 817, 825, 828
USER DATA, Group	Job Order 326
Statistics Screen 240 User ID	Prerequisite Condition 325
	Under ISPF 323
IOA Log Show Screen Window 304	Utilities Transfer Command
Online Facility 85	PF06/PF18 93
USER ID Parameter 130	
USER INTERFACE Option	V
Online Utilities Menu 323	V
USER Message Type	V Option
IOA Log Show Screen Window 303	Active Environment Screen 181, 234
User Plan	VALUE Attribute
User Prompting Plan 356	ODATE 65
User Profile	VALUE Field
Active Environment Screen Filter 190	Parameter Prompting 343, 347, 359
Customizing 81	
User Prompting Plan	Variable Assignment Definition 763
Parameter Prompting 355, 831	Variable Database
USER REPORTS Option	
Primary Option Menu 88	Updating 281
User-Defined Variable 754	Variable Database Facility 271
USER=Library	Adding Variables 277
MEMLIB Parameter 523	Database List Screen 273
OVERLIB Parameter 572	Row Numbering 277
USERID Field	Variable List Screen 274
Log Screen 297	Variable Zoom Screen 278
USERID Value	VARIABLE DATABASE Option
DO SHOUT Destination 470	IOA Primary Option Menu 87
SHOUT Parameter 629	Variable Database Option
UserDefined Variable	Primary Option Menu 87
AutoEdit 746	Variable Database, IOA 898
DO SET Statement 465, 747	Variable List Screen
SET VAR Parameter 466, 622	Variable Database Facility 274
Source Priority 765	Variable Member
USS	Format 759

Variable Resolution Concatenation 767	Active Environment Screen 189
	WAIT SCHEDULE (WITH RESTART) Status
Logic 767	Job Rerun 226
Variable Zoom Screen	WAIT SCHEDULE Field
Variable Database Facility 278	Global View Screen 200, 202, 204
Variables	WAIT SCHEDULE ON SPOOL Status
AutoEdit, Date, Global, KSL, Local, System, User-	Active Environment Screen 188
Defined 748	CMEM Forced Job 686
VAULT DEFINITION Option	WAIT SCHEDULE Status
Primary Option Menu 89	Cause or Reason 206
Version Information	CONTROL Resources 286
Primary Option Menu 90	Group Entity 566
VG Command	Job Deletion 211, 248
Active Environment Screen 180, 201	Job Graph 203, 205
VIEW Command	Job Rerun 226
Active Environment Screen 180, 199	Screen Status 188
VIEW GRAPH Command	Show Screen Filter 195
Active Environment Screen 180, 201	TERMINATE Option 247
View Graph Screen	WAIT SUB Status
Color Terminals 202	Show Screen Filter 195
Fields 202, 204	WAIT SUBMISSION Status
Format 202	Active Environment Screen 189
Group Statistics 201	WAIT_CONF Status
Non-color Terminals 204	CTMAPI 890
TOTAL Field 202, 204	WAIT_ORD Status
VIEW Option	CTMAPI 890
Active Environment Screen 181, 234	WAIT_PIPE Status
VIEWALL Command	CTMAPI 890
Job Order Execution History 236	WAIT_SCH Status
VOL=SER=%%VOLSER	CTMAPI 890
Example 803	Warning Message
VTAM 81	MEMNAME Parameter 528
Environment 95	TASKTYPE Parameter 654
PF06/PF18 93	WCAL Field
	WDAYS Parameter 381
NA/	WCAL Parameter
W	Calendar Name 670
	Calendar Type 672
WAIT CONFIRM Status	Non-periodic Calendar 670
Show Screen Filter 195	Periodic Calendar 672
WAIT CONFIRMATION (FOR SCHEDULE) Status	WDATE Parameter
Active Environment Screen 188	AutoEdit Simulation 331
WAIT CONFIRMATION (WITH RESTART) Status	WDAYS Parameter
Active Environment Screen 188	Description 669
WAIT CONFIRMATION Status	Example 673
CONFIRM Parameter 406	Format 670
WAIT EXEC Status	Logic 670, 672
Show Screen Filter 195	MINIMUM Parameter 531
WAIT EXECUTION Status	Negative Value 672
Active Environment Screen 188	PDS Parameter 579
WAIT FOR ODATE Field	Scheduling Logic 382
Scheduling Confirmation 153	
Zoom Screen 216	WCAL Field 381
WAIT FOR ODATE Parameter 327	WHEN Parameter
WAIT FOR ODATE Farameter 327 WAIT RELEASE Status	SHOUT Parameter 628, 631
Active Environment Screen 188	WHY Option
	Active Environment Screen 180, 205
WAIT SCHEDULE (PIPE) Status	Why Screen

Adding Conditions 208 Deleting Negative Conditions 208	Y
Deleting NOT-CONDs 208 Example 205	YEAR Field Calendar Definition 316
Reasons 206	Calendar Facility Entry Panel 309
WAIT SCHEDULE Status 205	Job Scheduling Plan Screen 165
Window Exit	Year List Screen 310
RESET Command 99	Commands 312
Windows	Exiting 321, 322
Active Environment Screen Delete 211	Inserting New Year 313
ADD Conditions 286 CMEM Rule Exit Option 264	Year List Screen, IOA Calendar Facility
CMEM Rule Exit Option 204 CMEM Rule Table Order 267	Format 311
CMEM Table Deletion 266	
Conditions/Resources Delete 287	<b>Z</b>
CONTROL-M/Restart	
Rerun Confirmation 225	Z Option
Delete Conditions/Resources 287	Active Environment Screen 180, 215
IOA Log Show Screen Window 301, 304	ZOOM Option
Manual Condition Add 293	Active Environment Screen 180
Manual Condition Delete 294	Zoom Screen
Overwrite Confirmation 364  Quick Schedule Definition Exit 368	Deleting PIPE Statements 821
Rerun Confirmation 222	Exiting 221 Fields 215
Resource Quantity 288	Job Order Information 215
Save Documentation 147	MAXRERUN Parameter 516
Scheduling Facility Exit Option 149	SHOUT Parameter 632
Show Screen Filter 192	
Why Screen Confirmation 209	
Zoom Screen Confirmation 221	
Wish WO0945 269	
WITH RESTART Field	
Restart Confirmation Window 226 WITH RESTART Status	
Active Environment Screen 189	
WO0943	
APPLY=YES 269	
Working Date	
Definition 63	
System Variable 750	
Working Days	
WDAYS Parameter 669	
WRN Task Type	
IOA Log Show Screen Window 304 Show Screen Filter 196	
Show Screen rater 190	
X	
X Command	
Online Facility Exit 91	

X Option

Primary Option Menu 87

### Notes



*91025*